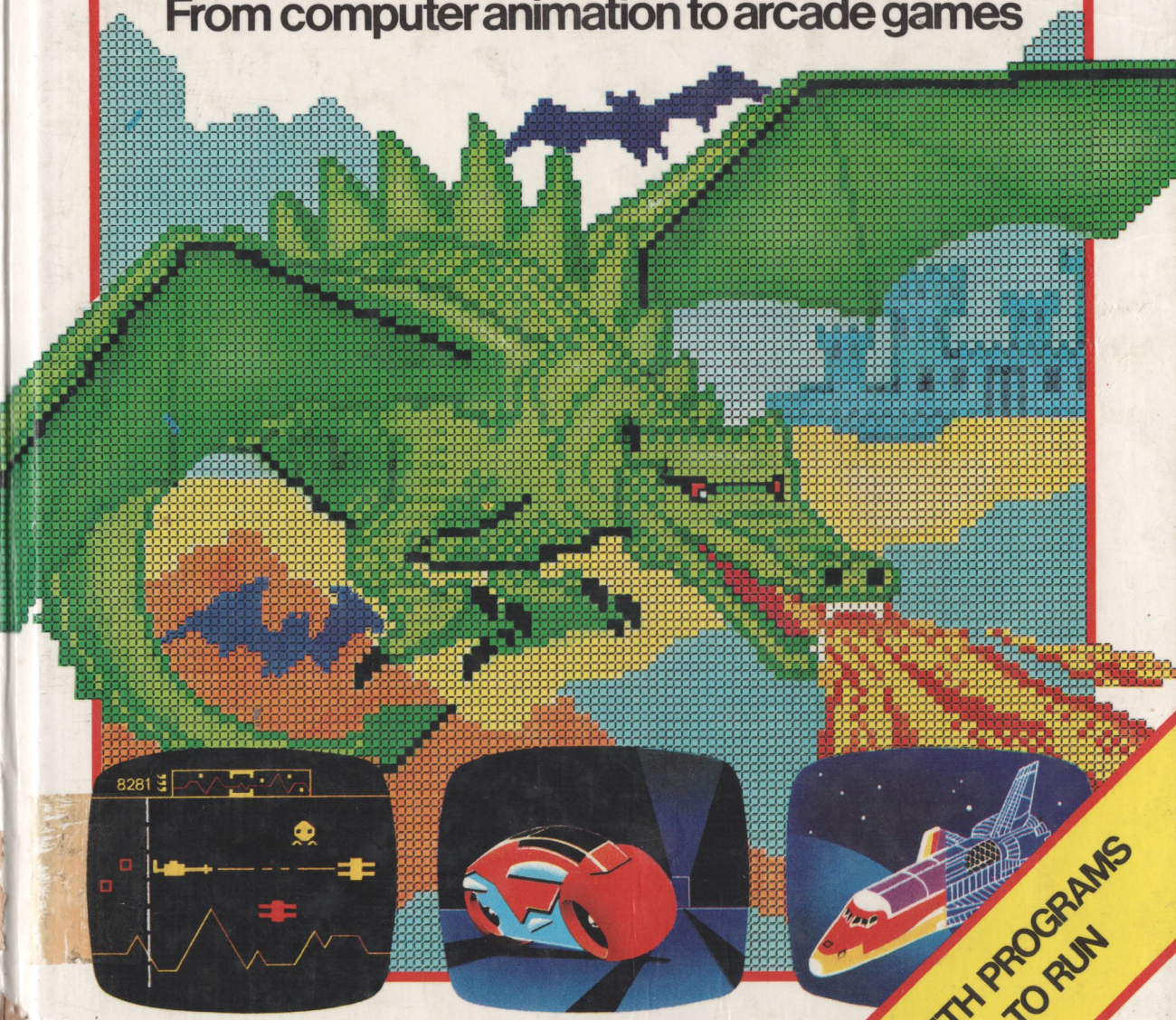
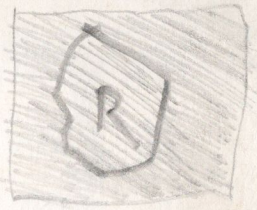
UNDERSTANDING
COMPUTER GRAPHICS

From computer animation to arcade games



WITH PROGRAMS TO RUN

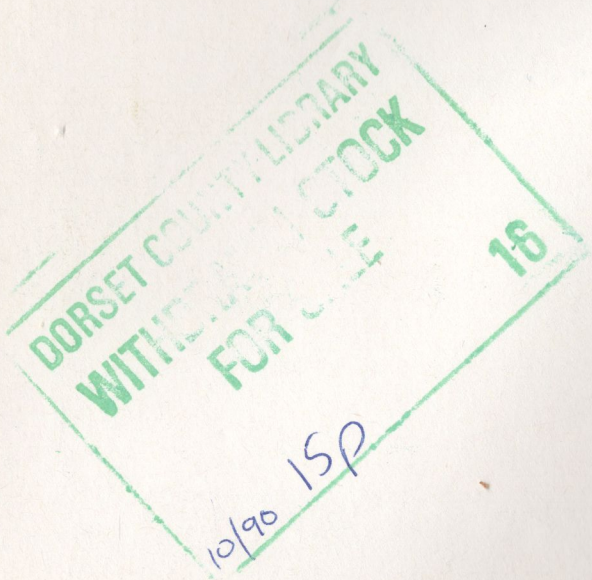
Usborne Computer Books



0860207404 014

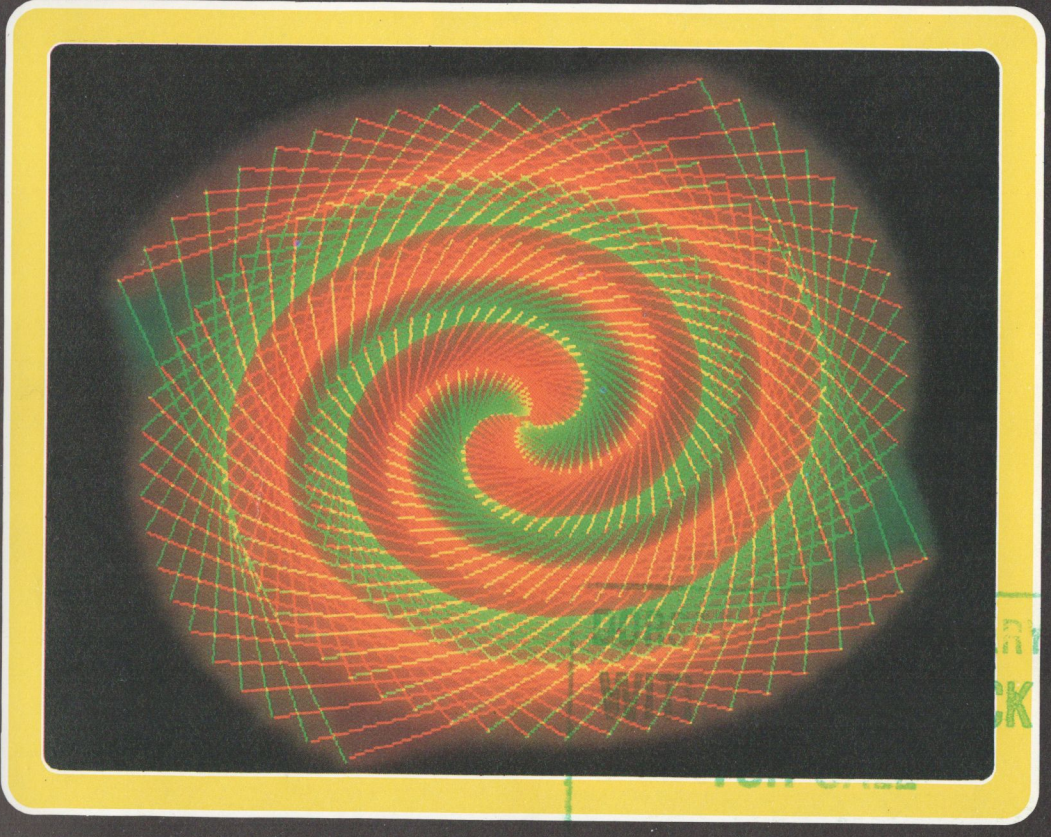
J001.6443
TATCHELL, JUDY UNDERSTANDING
COMPUTER GRAPHICS, BY J.TATCHELL &
L.HOWARTH 1983

N


DORSET COUNTY LIBRARY
WITHDRAWN FROM STOCK
FOR SALE
16
10/90 15p

UNDERSTANDING COMPUTER GRAPHICS

Judy Tatchell and Les Howarth



This book was designed by Graham Round and Roger Priddy and illustrated by Martin Newton, Mick Gillah, Graham Smith, Kuo Kang Chen, Graham Round, Rob McCaig, Naomi Reed and Projection Photographic Services.

Program consultant: Chris Oxlade

Contents

4 How a computer makes pictures
6 Pictures on a screen
10 Giving the computer graphics information
12 Games on the screen
14 3-D graphics
16 Computer-aided design
18 Computer simulation
20 Solving problems with graphics
22 Computers and special effects
24 TV graphics
26 Animated graphics
28 Displaying information
30 Computer art
32 Graphics programs to run
34 Pattern generator
36 In-betweening
38 Turtle graphics
41 Bouncing Bug
44 BASIC conversion chart
46 Graphics words
48 Index

DORSET COUNTY
02 APR 1984
LIBRARY HQ

About this book

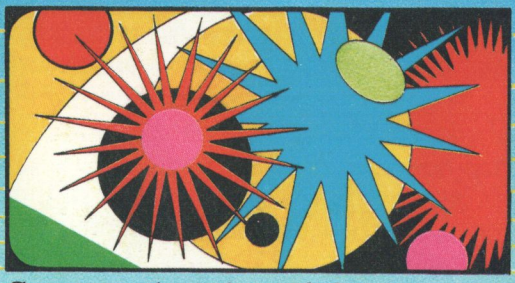
Computer graphics are everywhere: in arcade games, on home computers, in films and on television. You have probably seen lots of computer graphics without even realizing it. This book tells you about the ways in which computer graphics are useful as well as being exciting to look at. You can find out how information fed into a computer turns into a picture on the screen and there are some programs to run on a home computer* so you can create your own computer graphics.

Here are some examples of computer graphics. You can read about how they are done later in the book.

Special effects

A computer-generated background can be used with film of real actors to create an imaginary world.

Computer art

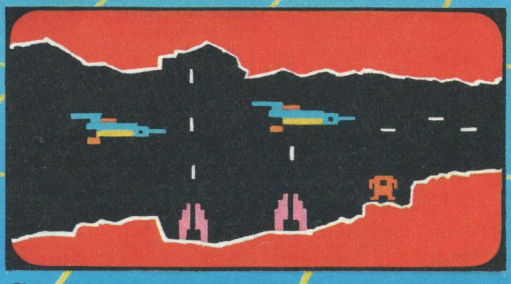


Computers give artists a whole new medium to work in. They can make pictures electronically without using paint and brushes.

Cartoon animation

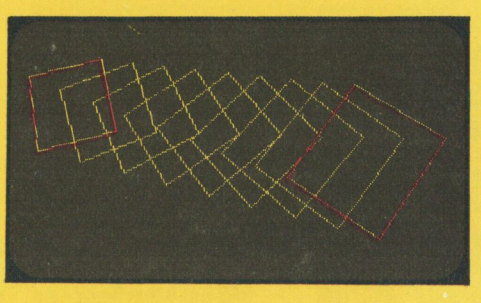
Cartoons can be drawn directly on a computer screen and the computer itself can even generate some of the pictures.

Arcade games



Computers create fantastic games. You can find out how they do this and how they make characters in the game move.

Graphics programs to run



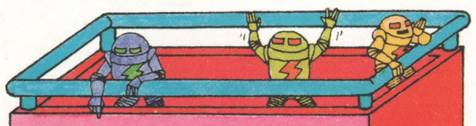
In the last section of the book are four graphics programs for you to try on a home computer. They will run on the BBC, Sinclair Spectrum (Timex 2000), Dragon and Apple computers. You can see for yourself how a computer does some of the things explained earlier in the book, such as changing one shape into another. Remember, though, that the graphics will not be as spectacular on a home computer as those produced by more powerful computers.

*Programs will run on BBC, Sinclair Spectrum (Timex 2000), Dragon and Apple computers.

How a computer makes pictures

To make pictures, or graphics, a computer needs to be given a list of instructions called a program, telling it where to draw lines, what colours to use, and so on. These pages show you what happens to the program inside the computer and how a picture appears on the screen. Programs are also known as software. The computer itself and equipment that is attached to it, such as the screen, are known as hardware.

You can type in a graphics program on the computer's keyboard, or you can use special graphics equipment to feed in information for a picture. You can find out more about this kind of equipment on pages 10 and 11.



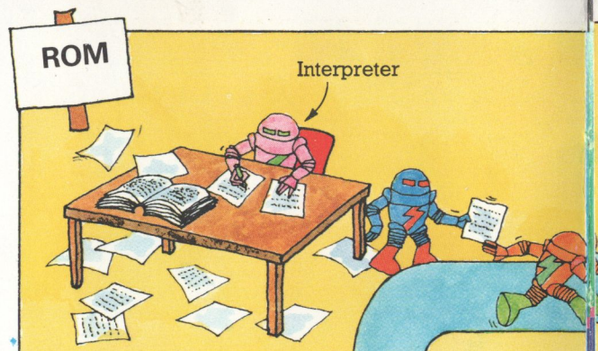
CPU

The central processing unit (CPU) controls the workings of the computer. When you type in a program, the CPU stores it in the memory (see below) and then carries out the instructions.

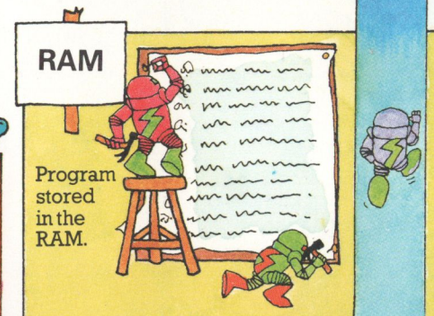
Follow the arrows to find out what happens inside the computer.

The computer's memory

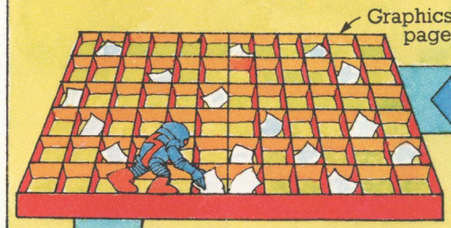
The computer has two kinds of memory. The random access memory (RAM) stores temporary information that can be changed or wiped out, such as the program. The read only memory (ROM) stores instructions the computer needs all the time to tell it how to operate. These are permanent and cannot be wiped out.



When the program is run, the interpreter in the ROM translates it line by line into the computer's own code, called machine code.

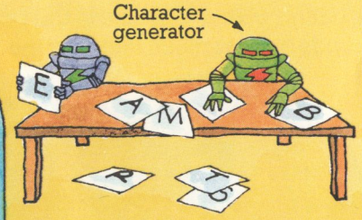


The program is stored in the computer's random access memory, or RAM. You can use, or run, the program as many times as you like until you give the computer an instruction telling it to clear its RAM. Switching the computer off also clears the RAM.

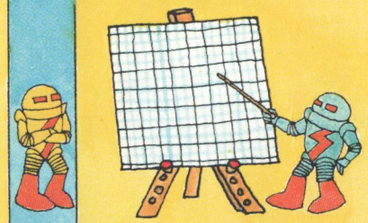


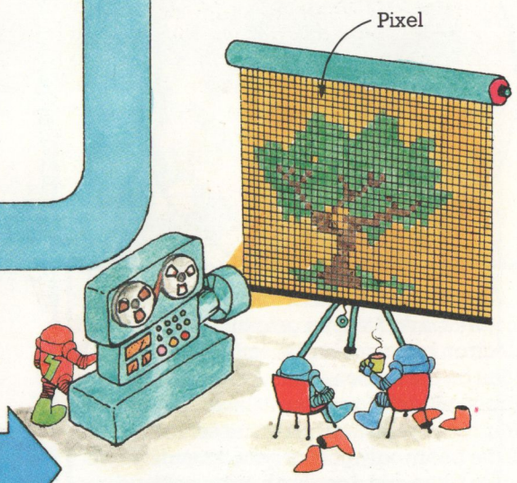
The area in the RAM where information about the pixels is stored is called the graphics page. Each pixel has a separate place in the graphics page with a number address. Information for the pixel is stored as another number representing colour and brightness.

If the program contains text to be printed on the screen, a special program in the computer's ROM called the character generator tells the computer what shapes to light up to make letters, symbols or numbers.



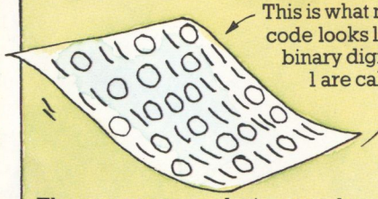
The computer divides the screen into tiny areas called pixels and turns each one on or off to make shapes. A special area in the ROM tells the computer where the information about each pixel is stored in the RAM.



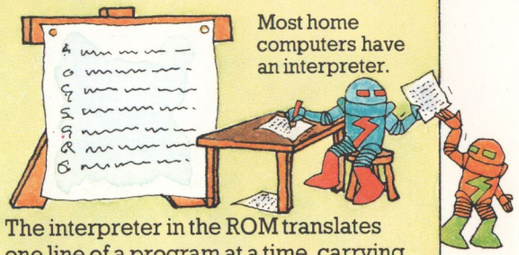


The computer turns each pixel on or off and makes them different colours according to the information about them stored in the graphics page in the RAM.

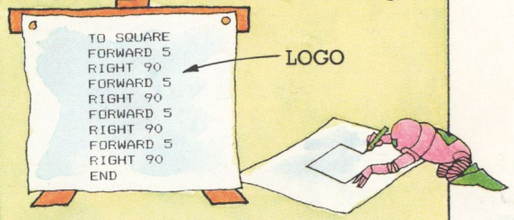
Programming languages



The computer works in a number code called machine code. This is a binary code made up of different combinations of the digits 0 and 1. Most programs are written in easier programming languages consisting of words, numbers and symbols. These are translated into machine code by a permanent program in the ROM called an interpreter.



The interpreter in the ROM translates one line of a program at a time, carrying out the commands in that line before translating the next. Some computers have compilers instead, which translate the whole program before carrying out any of the instructions. The program then runs much faster. Special graphics computers have compilers for programs with lots of movement or colour changes.

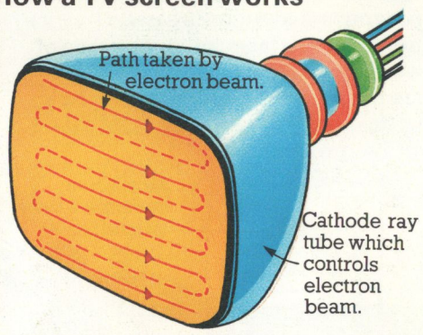


Some programming languages, such as LOGO, are specially designed for graphics. Most home computers understand BASIC. The BASIC graphics commands vary a lot between computers, though. There are four graphics programs in BASIC at the back of this book with instructions for converting the commands for different computers.

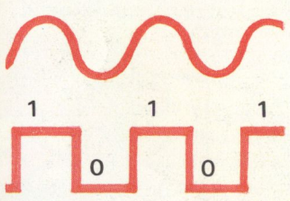
Pictures on a screen

A home computer usually uses an ordinary television for a screen display but there are also screens called monitors which are specially designed for computers. These give clearer pictures. You can find out about the difference between TVs and monitors below. If you look closely at a picture on TV you will see that it is made up of tiny dots lit up in different colours. Most computers cannot control each dot individually, so they deal with them in groups, called pixels. The size of the pixels determines the quality of the picture and depends on how much memory the computer has to store information about them.

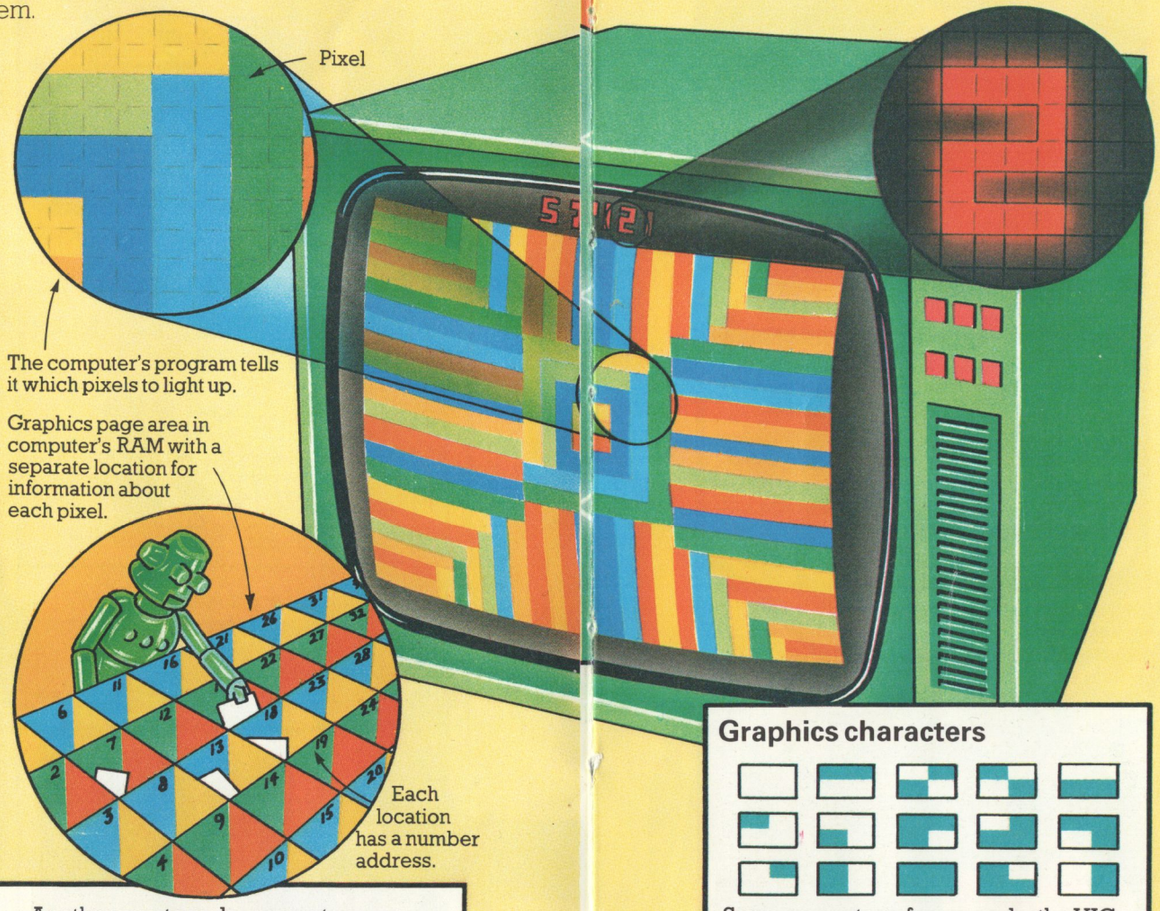
How a TV screen works



The main part inside a television is called a cathode ray tube (CRT). An electron beam is fired through the CRT to scan the back of the screen. It weaves its way across and down the screen hundreds of times a second, faster than the eye can see. On its way, it lights up various dots on the screen to make the picture. This is called "raster scanning".



A TV is designed to receive analogue signals, that is, continuously varying waves, from the broadcasting station. A computer sends digital signals to the TV which are square waves representing the binary numbers the computer works in. The TV rounds off the edges of the waves so the picture on the screen is slightly blurred. A monitor is designed to receive digital signals so it gives a sharper picture.


Pixel
The computer's program tells it which pixels to light up.
Graphics page area in computer's RAM with a separate location for information about each pixel.
Each location has a number address.

Another way to make computer pictures is with vector graphics. The electron beam moves directly from one point on the screen to another in any direction leaving a straight line. Vector graphics are faster than raster scanning as the beam only goes where it needs to make a line and does not have to scan the whole screen. The Vectrex computer games module uses vector graphics. It has its own small screen.

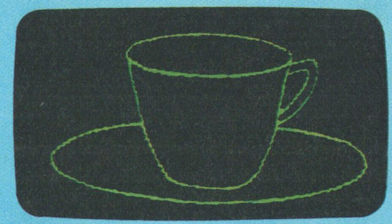
The character generator

The character generator in the computer's ROM is a permanent program telling it which dots to light up to show letters, numbers, punctuation and symbols on the screen. The dots making up characters are usually smaller than the pixels used to make graphics.

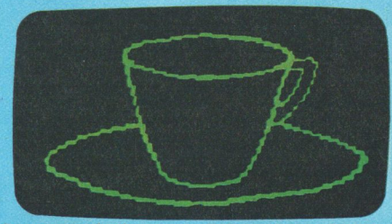
Graphics characters

Some computers, for example, the VIC 20, Sinclair Spectrum and ZX81 (Timex 2000 and 1000), can put pre-set graphics characters on the screen. These are simple shapes made by the character generator in the same way as other characters, by lighting up combinations of dots. They are printed on keys on the keyboard and are automatically displayed on the screen when you press those keys.

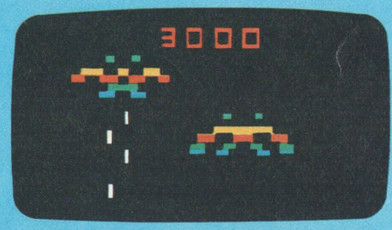
Resolution, modes and memory



Different computers divide the screen into different numbers of pixels, depending on the size of their RAM. Pictures with thousands of very small pixels have fine lines and smooth curves. These are called high resolution graphics and need a lot of memory.



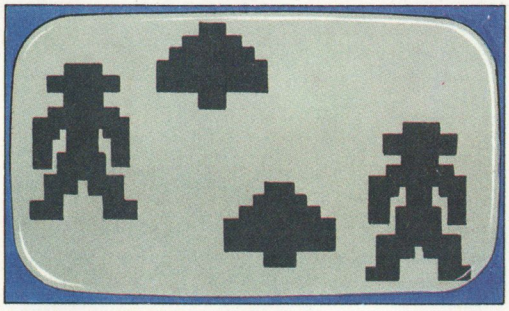
Low resolution graphics are made up from fewer, larger pixels which take up less memory. Lines are thicker and curves more "stepped". The number of colours the computer can use also depends on the amount of memory available to store the information.



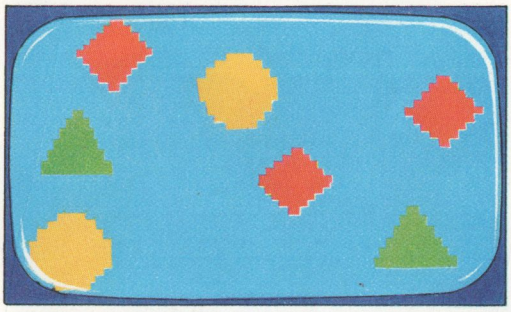
Some computers give you the choice of several graphics modes. Each mode has a different resolution and number of colours. Computers that do this usually have a special graphics chip to control the screen display. This means that the computer can produce graphics much faster, which is useful for moving pictures or complicated designs.

More about resolution and memory

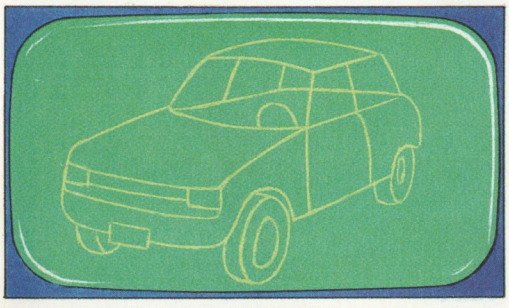
These pictures show what different resolutions and numbers of colours look like and how much memory they need. Computer memory size is measured in kilobytes (K). There are eight bits (binary digits) in one byte and 1024 bytes in a kilobyte. High resolution graphics in lots of colours use the most memory. Definitions of high and low resolutions keep changing as computers become more powerful and better resolutions become standard.



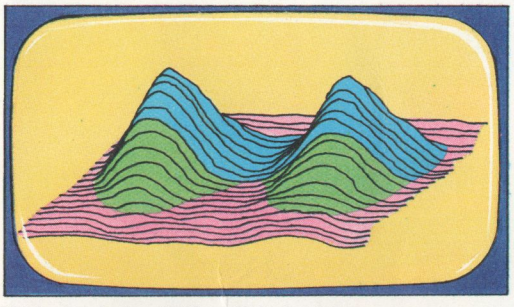
Very low resolution (63 × 43 pixels)
Black and white
1K memory



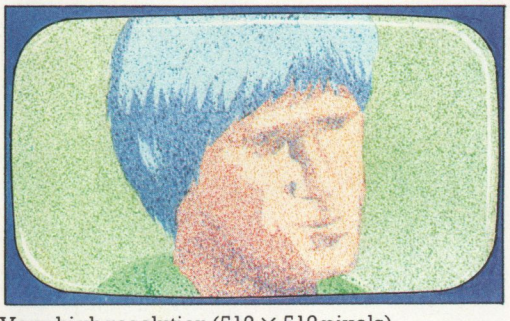
Low resolution (160 × 256 pixels)
4 colours
10K memory



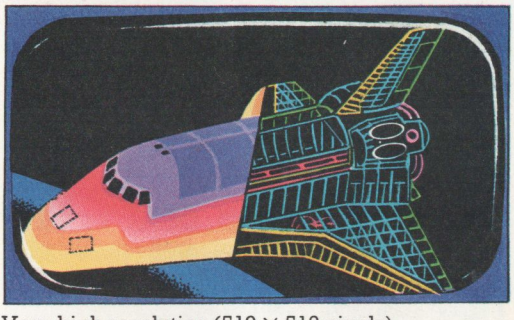
High resolution (640 × 256 pixels)
2 colours
20K memory



High resolution (640 × 256 pixels)
4 colours
40K memory



Very high resolution (512 × 512 pixels)
8 colours
96K memory



Very high resolution (512 × 512 pixels)
16 colours
128K memory

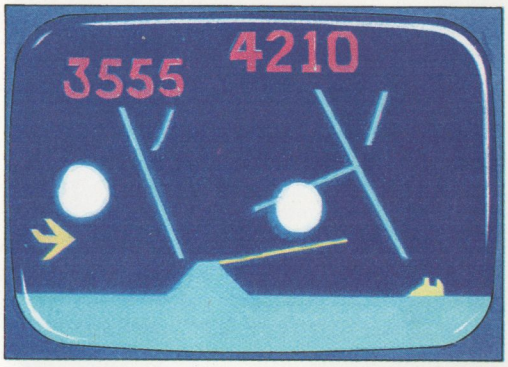
The pixels in the very high resolution displays are almost as small as the dots in the picture of a broadcast TV programme. However, to get the same picture quality you would need at least 255 colours, which would take up 256K memory.

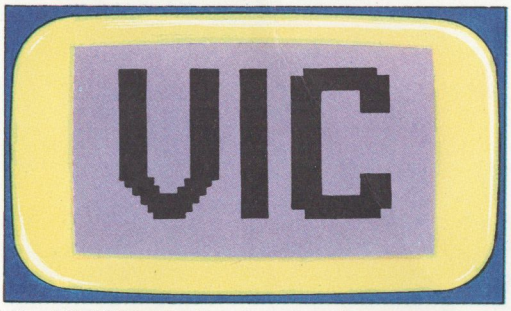
You need a colour monitor instead of a TV to display very high resolution graphics, or you will lose some of the detail in the picture. See page 6 to find out how a monitor differs from a TV set.

Home computer graphics

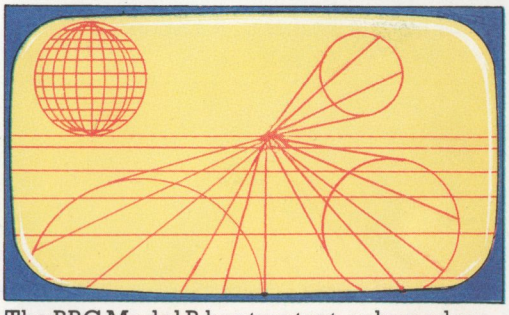
Here are screen displays of three home computers showing how they use resolution and colour. Most home computers come with 8K or 16K RAM, but you can usually add more memory up to at least 64K.

The Atari 800 has ten modes with resolutions of up to 320 × 192 pixels. It has up to 16 colours each with 16 different intensities and can make exciting graphics for computer games.



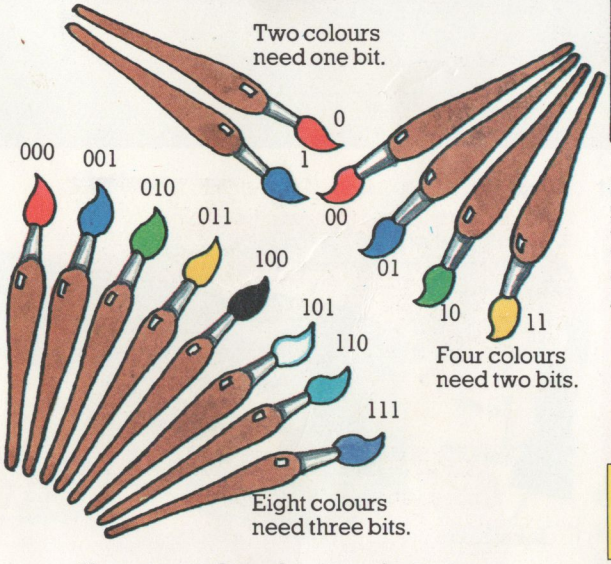


The VIC 20 has a border round a screen area for text and graphics characters. You can choose from eight colours for the border and text and from 16 colours for the screen area.

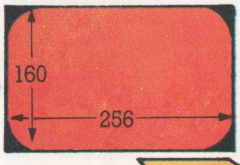


The BBC Model B has two text-only modes and five graphics modes with different resolutions. In the very high resolution mode (640 × 256 pixels) you can only use two colours. In other modes you can use two, four or sixteen colours.

Working out memory size



You can work out how much memory a computer needs for any resolution and number of colours. Each colour is controlled by a bit. The bit can be 0 or 1, so a single bit can control two colours. Two bits control four colours, three bits eight colours, and so on.



1. Multiply pixels across by pixels down.

160 × 256 = 40,960

2. Multiply by number of bits needed to control colours. This gives answer in bits.

40,960 × 2 = 81,920

Two bits needed to control four colours.

81,920 ÷ 8 = 10,240

10,240 ÷ 1,024 = 10

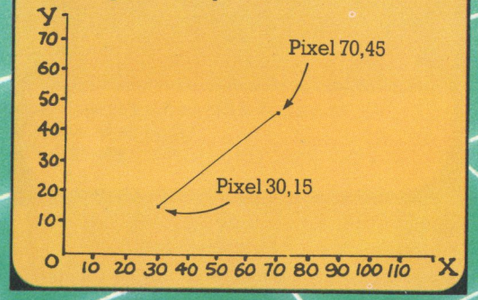
3. Divide by eight for answer in bytes and again by 1,024 for answer in kilobytes.

To work out how much memory the computer needs, you multiply the number of pixels on the screen by the number of bits needed to control the colours and then divide, as shown above, for the answer in bytes or kilobytes.

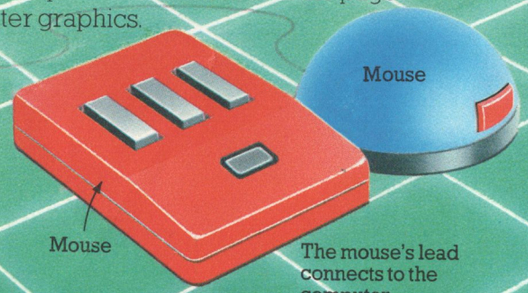
Giving the computer graphics information

There are lots of different ways to put information for pictures into a computer. You can type out a program on the keyboard or you can use special pieces of equipment like those on this page. They are easy to use as they feed information about lines, colours and shapes into the computer automatically. This kind of information is analogue (i.e. it can vary continuously) so the computer needs an analogue-to-digital converter, which is some extra electronic circuitry, to turn the information into numbers. The computer also needs a program telling it how to use the information. Equipment like this is used commercially but you can buy cheaper versions for a home computer. At the bottom of the page are some different ways to display computer graphics.

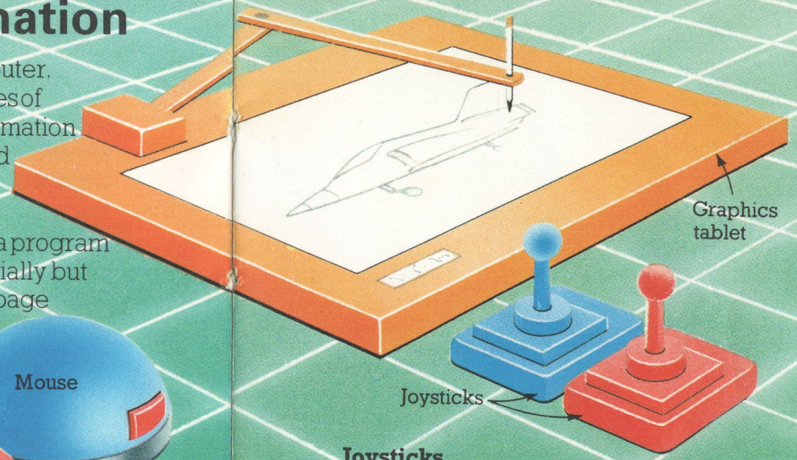
Using the keyboard



The pixels on the screen are arranged in rows and columns. To draw a line for a picture, you type in the X and Y co-ordinates of two pixels and tell the computer to join them up, using a graphics command such as DRAW or LINE.



Mouse
A mouse has wheels underneath and as you push it over a flat surface, the computer registers distance and each change of direction and draws a corresponding line on the screen. It is useful for tracing shapes and has buttons on it to give the computer extra commands such as when to stop and start drawing.

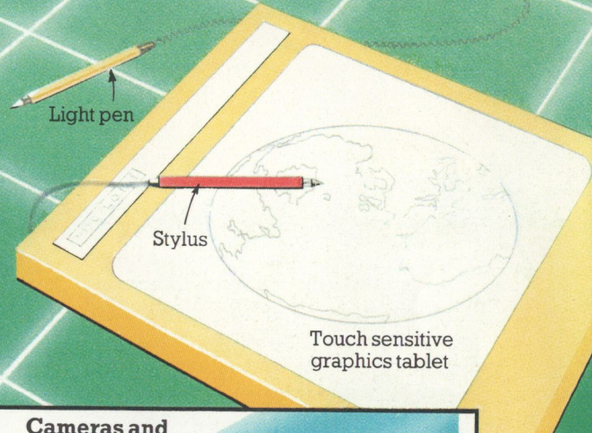


Joysticks
Some games joysticks can also be used for drawing pictures. The computer needs a program telling it to draw a line on the screen in the same direction as you move the joystick.

Light pen
You can draw on the screen with a light pen which has a light-sensitive tip. When it is placed on the screen, it senses the TV's electron beam as it flashes past. It sends a signal to the computer which can work out the position of the pen because it knows where the electron beam is at any time. The computer lights up the pixels the pen passes over to make lines.

Graphics tablets
This type of graphics tablet has a pen (or another kind of pointer) supported in a jointed arm. Electronic components called potentiometers measure the angles in the joints as you move the pen. The computer then calculates the pen's position and draws corresponding lines on the screen.

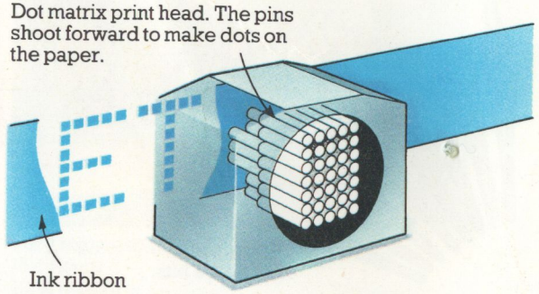
If you draw with a special pen, called a stylus, on a touch-sensitive graphics tablet or digitizing pad, the picture appears on the screen. The tablet reads each position of the pen as a point with X and Y co-ordinates. The computer's program converts these readings into co-ordinates for the screen. This sort of graphics tablet is more accurate but also more expensive than the other sort.



Displaying graphics

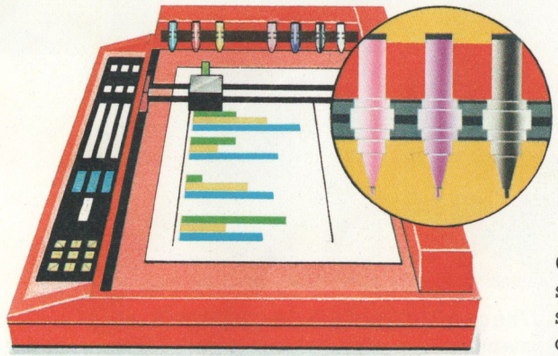
A monitor can be either monochrome (black and white) or RGB (colour). RGB stands for red, green and blue. TVs have RGB screens, too, and different combinations of these colour dots make all the colours you see in the picture.

Printers



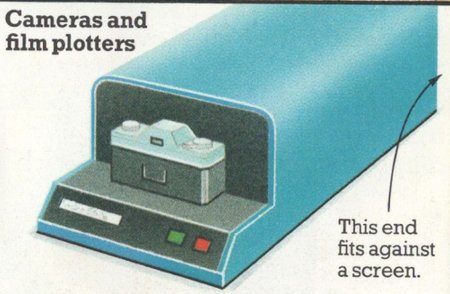
Dot matrix printers have a group of little pins which hammer out dots to make letter shapes. Some printers can control these pins individually and print out graphics. Each dot is equivalent to a pixel on the screen.

Plotters



A plotter uses pens supported over a sheet of paper. The computer tells it which colour pen to select and where to move it. Some plotters are very big and give enlarged copies of computer graphics.

Cameras and film plotters



Graphics can be photographed with a special camera from a flat, non-glare screen. The camera takes three exposures, one each for the red, green and blue signals.

Expensive film plotters record graphics straight from a computer on to film using fibre optics or lasers to trace the pattern. The results can be projected on to large wall screens.

Games on the screen

Many breakthroughs in computer graphics have resulted from fierce competition in the arcade games industry to produce more exciting visual effects for games. These pages show how some of the effects are created, using an imaginary game called Electro-Destroy as an example.

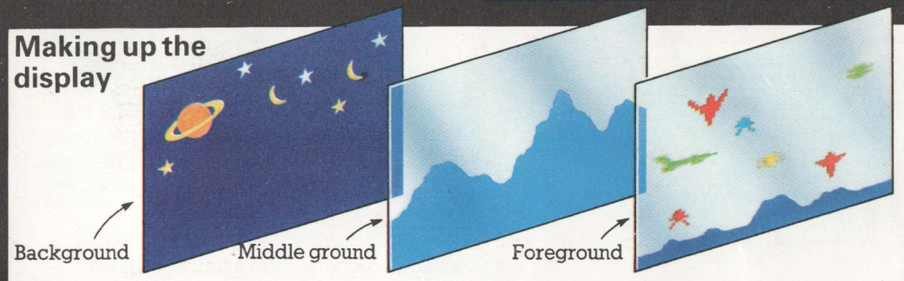
An arcade game contains a computer with a special "display processor" chip to handle the graphics. It processes images very fast so you get vivid explosions and colours, and objects or shapes can zoom around at speed in outer space or an alien landscape. Some home computers can be programmed to produce similar effects, especially those with special graphics chips such as the BBC, Atari, VIC 20 and Commodore 64.

Windows

The screen display in the game is divided up into different areas, or windows, for graphics and text. The position and size of a window is defined in the game program.

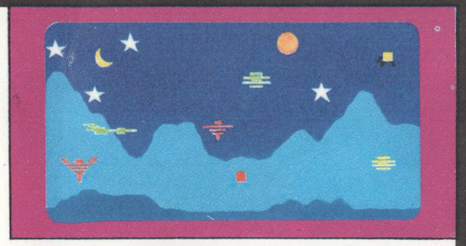
In the picture on the right, the text window at the top of the screen shows the title and score. A small graphics window superimposed on the text window shows a "radar" picture of the enemy fleet's position. The game "playfield" where the action takes place is displayed in a large graphics window across the rest of the screen.

Making up the display



The main display consists of several layers of scenery. You can imagine the layers as sheets of glass each painted with different parts of the scene, one in front of the other. Each layer has its own separate area, or graphics page, in the computer's memory for storing details.

The graphics pages in the computer's memory have priorities assigned to them so that if two objects on different layers overlap, the computer knows which should come in front of the other. This gives the effect of foreground, middle ground and background.

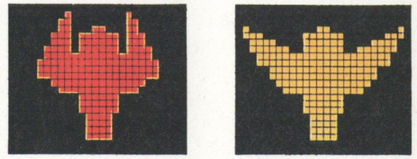


In this game, the landscape moves from right to left so that it looks as if the jet is flying forwards. The background layers of scenery move more slowly than the foreground to give an impression of depth and perspective.

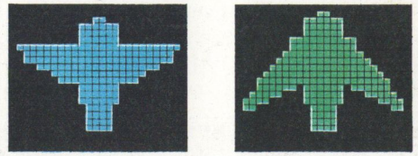
Electro-Destroy

You are a lone fighter pilot defending vital resources of fuel on your planet against an invading force of mutant aliens and birdmen. You must destroy them before they reach the ground, steal the fuel and double in strength. You are also under attack from the enemy fleet of deadly flying saucers which swoop in unexpectedly.

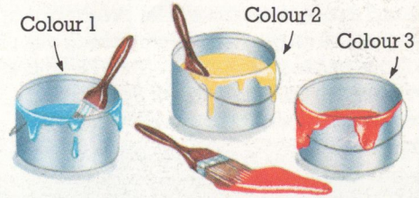
Animation and colour-swapping



Four birdman positions



The birdmen in the picture flap their wings as they move down the screen. This effect is made by storing four versions of the character in different positions and displaying different versions as the birdmen descend.



The computer controlling the game has a number-coded "palette" of colours which it can use on the screen. It can change the colours in the picture by instructing, for example, "change everything in colour 1 to colour 3". This is called colour-swapping. In some games, the characters have their own palettes so that they can change colour without affecting the rest of the picture. Changing their colour can show their strength or register hits.

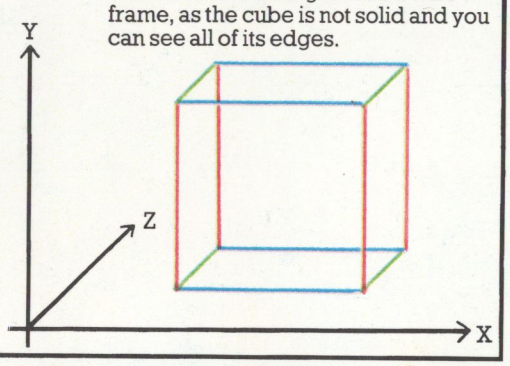
3-D graphics

Three-dimensional objects have length, width and depth. Artists give an impression of depth to a picture on a flat piece of paper by drawing lines in perspective. These depth lines appear to travel into the picture. A computer needs to be told the length and angle of depth lines, or it can be given a program telling it how to work them out. 3-D graphics programs run quite slowly on a home computer because of all the calculations involved. Extremely powerful computers can create stunning 3-D graphics, though. For example, they are used by NASA* to generate films which simulate space travel.

Drawing a 3-D picture

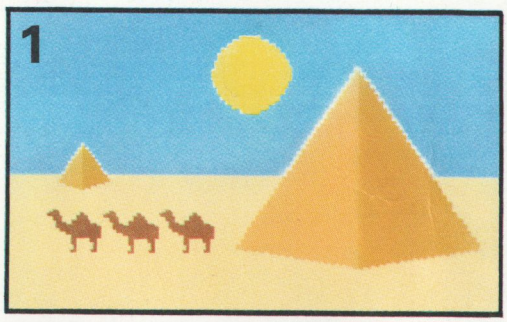
The cube in the diagram has 12 edges. The front and back edges are drawn parallel to the X and Y axes and the depth lines going into the picture are drawn parallel to the Z axis. If the cube is rotated or moved, all the lines and angles have to be calculated again and redrawn for each new position. The computer can be given rules to follow in a program so that the cube can spin, for example, and still keep its shape.

This kind of drawing is called wire frame, as the cube is not solid and you can see all of its edges.

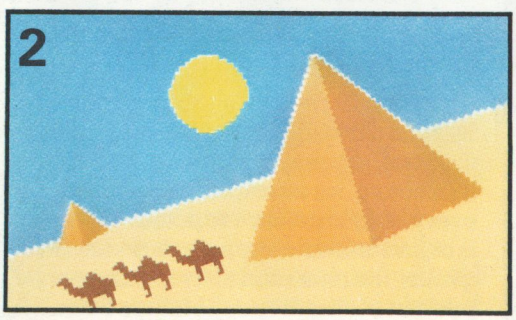


Changing your viewpoint

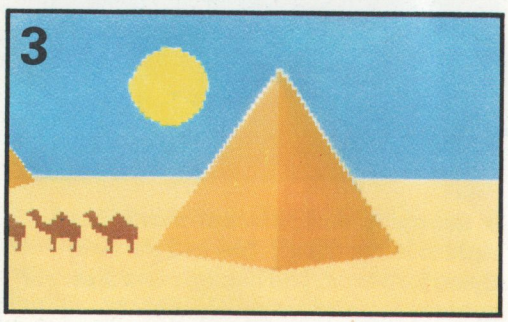
Some 3-D graphics programs can change the angle at which you look at a picture (your viewpoint). Instead of you moving, the computer moves everything on the screen in relation to you. This is called translation. It can even give the impression that you are travelling through a landscape, as in the Atari *Battlezone* game. The program works out which lines are visible, rubbing out, or clipping, those which would be behind you or out of sight. The ways in which your viewpoint can be altered are described here.



Pitch tells the computer whether you are looking up or down at the picture and by how much.



Bank describes the amount of tilting clockwise or anti-clockwise.



Heading tells the computer which direction you are looking in: north, south, east, west or anything in between.

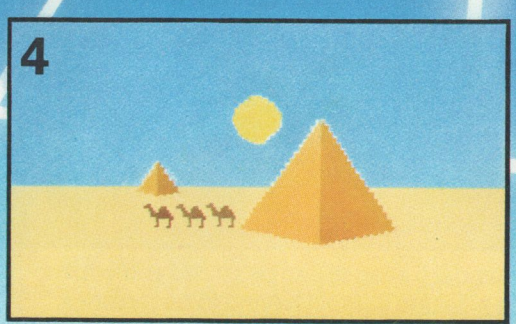
*North American Space Administration.

Solids and colour

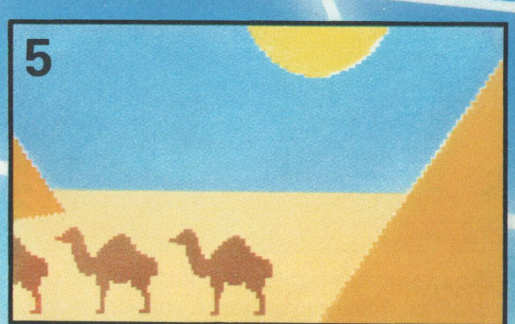
The computer plots curves by breaking them down into tiny straight lines. Some programs have a built-in smoothing routine to round off jagged edges.

Very high resolution, 3-D graphics such as these can only be created on a powerful computer with lots of colours and shades. The picture would first be drawn as a wire-frame drawing showing all the lines making up the objects and background. The computer's program can then make the objects look solid by removing any lines that go behind them. This is called hidden-line removal.

The impression of shape (for instance, the roundness of the moons) can be created by shading. The effect of depth and distance is increased by fading a colour according to how far away it is. This is called depth clueing.



A **wide-angle** view takes in an extra-wide sweep of the picture.



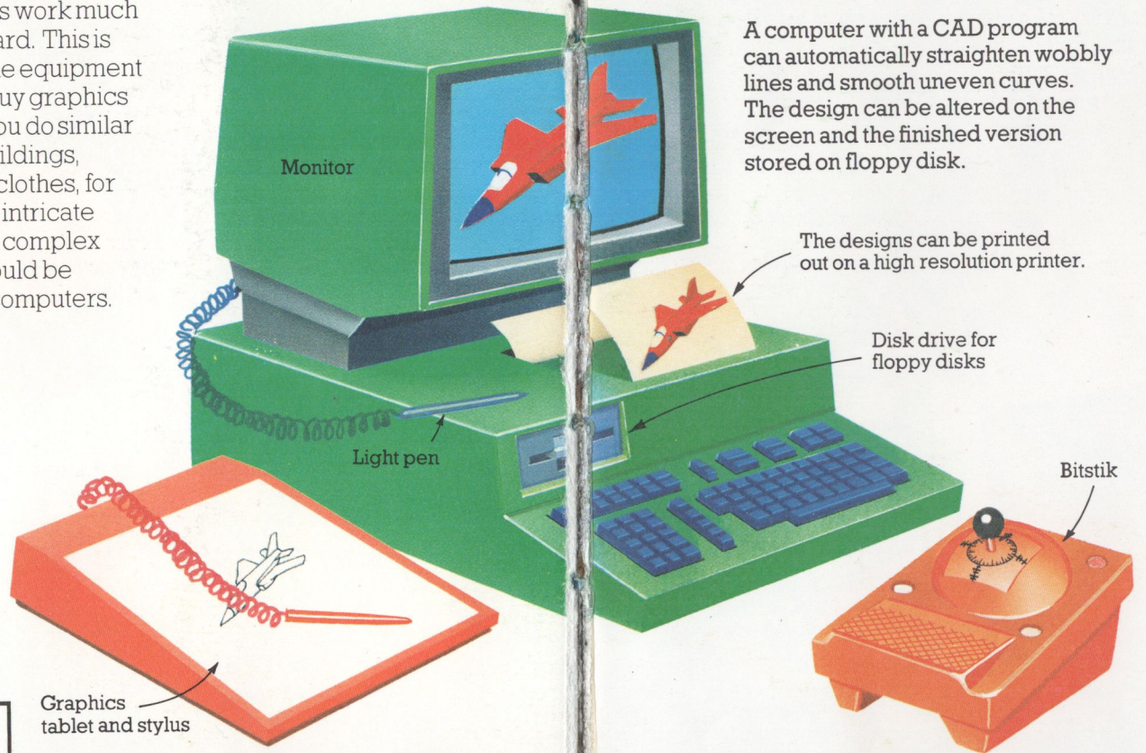
Telephoto view lets you zoom up closely to something in the distance.

Computer-aided design

Computers can help professional designers work much faster than with a conventional drawing board. This is called computer-aided design, or CAD. The equipment and programs are expensive but you can buy graphics programs for a home computer which let you do similar things. In industry, CAD is used for cars, buildings, aircraft, town plans, electrical circuits and clothes, for example. It is used in the design of tiny and intricate microprocessor chips, as well as large and complex machinery like the Space Shuttle which would be extremely complicated to design without computers.

CAD equipment

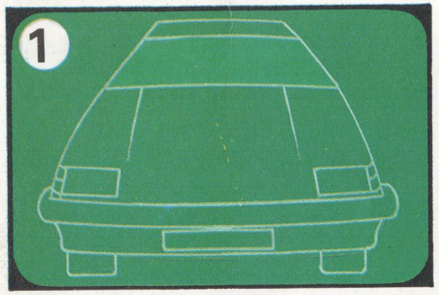
The designer uses a light pen or a graphics tablet to draw on the screen. He or she may work on a microcomputer, or a graphics terminal which is a screen and keyboard unit connected to a more powerful computer. The computer has a special CAD program which puts a list of choices on the screen. The designer can choose whether to draw, rotate the drawing, or enlarge part of it, for example. The designer selects one by touching it with the pen.



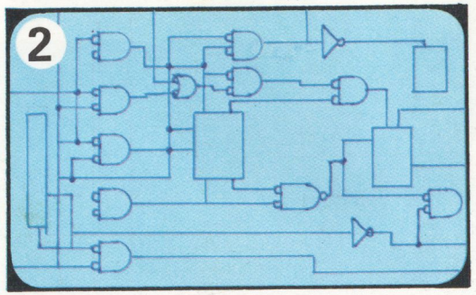
The computer can also be programmed to carry out calculations related to the design, for instance, how much material would be needed and what it would cost. For something like a house or a bridge, it can calculate stresses and work out whether the design is strong enough.

How computers save time

A CAD program can take over time-consuming work, like drawing an identical part several times or drawing the same object from different angles. The computer can instantly flash up a design stored in its memory for the designer to alter or update. Here are some other ways in which CAD saves time.



If a shape is symmetrical, such as the body of a car, the designer need only draw one side, and the computer can draw a mirror image on the other side to complete it.



Standard shapes can be stored and added to a design without having to be redrawn. For instance, symbols for electrical components can be positioned in a circuit design on the screen.

A computer with a CAD program can automatically straighten wobbly lines and smooth uneven curves. The design can be altered on the screen and the finished version stored on floppy disk.

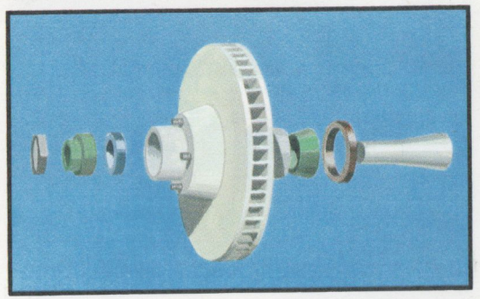
The designs can be printed out on a high resolution printer.

Disk drive for floppy disks

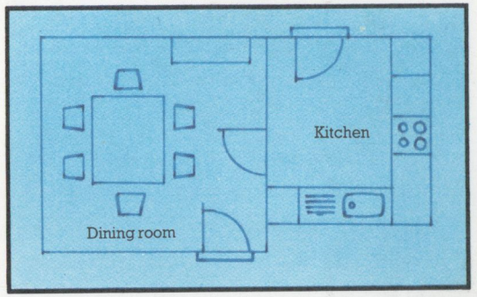
Bitstik

This piece of CAD equipment is called a Bitstik. It has a joystick to draw lines or move shapes and buttons for choosing what you want to do from a list on the screen. Twisting the knob on the joystick enlarges or shrinks a design to whatever size you want.

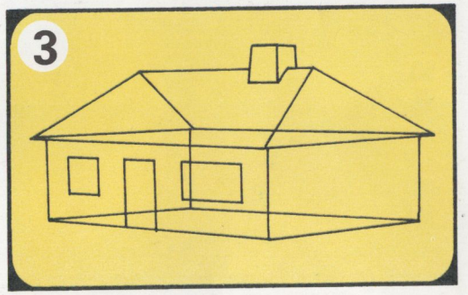
Other CAD features



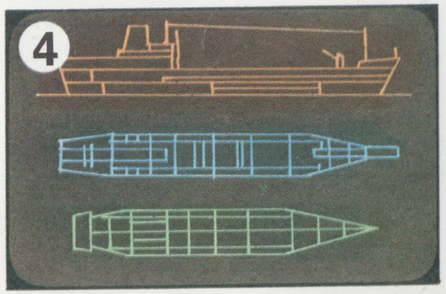
This piece of machinery would first be designed as a wire frame, or line drawing. The computer can then convert it to a solid using hidden line removal, colour and shading. The design can be rotated and viewed from other angles, or all the pieces fitted together to show how it works.



A designer can also zoom in on a section of a design, such as the floor plan of a house, to work more closely on part of it. The designer can experiment with different ways of positioning the kitchen units or doors.



An architect can draw a single view, or elevation, of a building and the computer carries out calculations to produce different elevations from the same set of measurements.



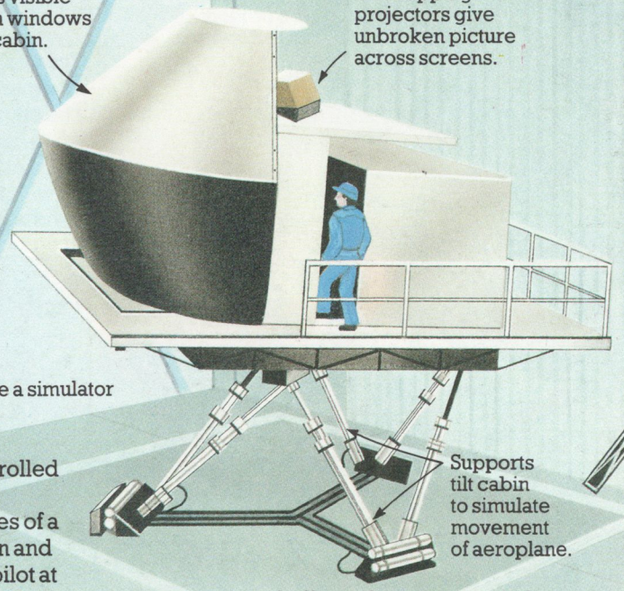
Designs produced on a computer can be transmitted around the world by cable or satellite. They can be sent directly to production engineers in factories who can then begin to manufacture the product.

Computer simulation

Computers can imitate real situations on the screen. This is called simulation and it is used in training programmes, or to carry out mock-up experiments which might otherwise be dangerous or very expensive. In simulations the computer works in real-time, that is, it responds directly to commands. You can get simulation programs for a home computer, for instance, versions of the commercial flight simulator on this page, or space travel games where you have to dock with another craft. Here are some examples of simulation.

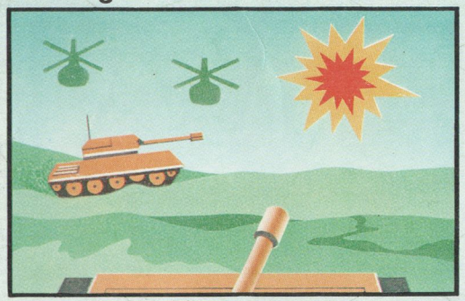
Flight simulator

Inside a simulator


Screens visible through windows inside cabin.
Overlapping projectors give unbroken picture across screens.
Supports tilt cabin to simulate movement of aeroplane.

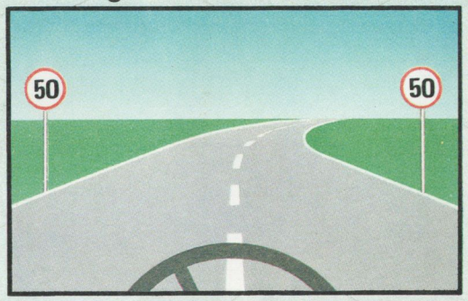
A flight simulator is a computer-controlled model of an aircraft flight deck. The computer projects moving 3-D images of a runway on to screens round the cabin and responds in real-time to the trainee pilot at the controls. It controls supports which move the cabin to simulate the feeling of taking-off, banking and landing, and generates sound effects. It can recreate the conditions of different airports at any time of day or night and in any weather.

Training tank crews



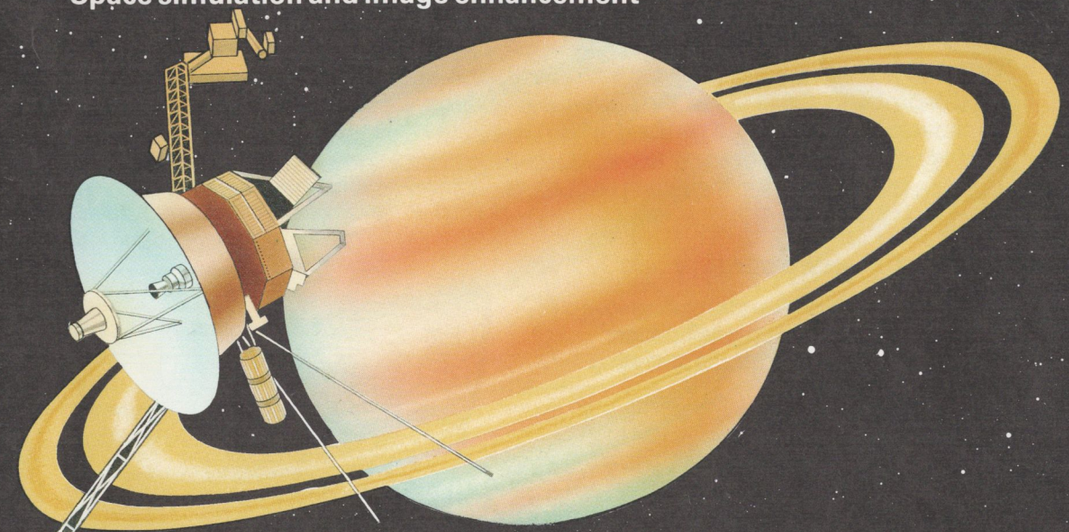
In the United States, a program similar to the Atari *Battlezone* game is used to train tank crews in simulated situations involving enemy tanks and helicopters.

Learning to drive



Systems using video disks linked to a computer can be used to familiarize learner drivers with cars and driving situations before they go on the road in a real car.

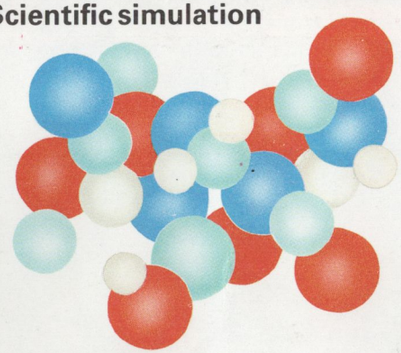
Space simulation and image enhancement



Picture information is sent back from space probes such as Voyager to computers at NASA. The signals can suffer from interference, similar to a long-distance phone call, so a process called image enhancement is used to filter it out. This produces clearer pictures by comparing each of the several million pixels making up the image with its neighbour and deciding which are true picture details and which are interference.

The computers also receive information about the performance of the spacecraft, which is sent to designers and engineers working on new craft. NASA scientists can put all the information together to make computer simulations of flights for the future, deciding where to steer the craft and direct its cameras. The very powerful computers used for these simulations have compilers* instead of interpreters to translate the program so that graphics can be generated fast enough to give a realistic impression of movement.

Scientific simulation



In chemistry and medicine, 3-D computer simulations are used to show the structure of molecules and how they behave. In the manufacture of drugs, for example, an accurate simulation can show whether the molecules of one chemical will join with those of a toxic substance to make it harmless.

Home computer simulation

```
DATE: 1/1/84
TIME: 2300
LATITUDE: 51.5
LONGITUDE: 0
```

This is an astronomy program for a home computer which simulates the sky. You type in any date and the latitude and longitude of anywhere in the world and the computer shows you the stars and planets that would be visible without a telescope.

*Compilers and interpreters are explained on page 5.

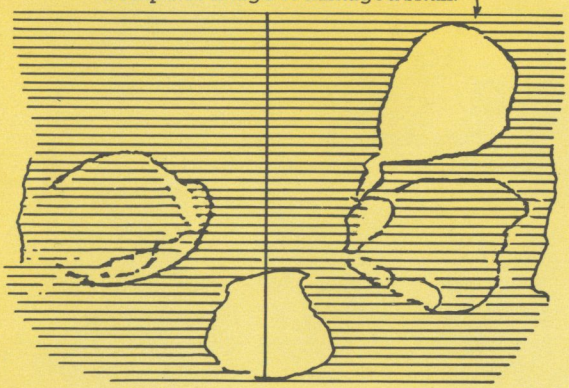
Solving problems with graphics

Once a computer has been told in a program what steps to take to solve a particular kind of problem, it can follow the same instructions to work on different sets of data. Computer simulations can be used to help work out how to handle a critical real-life situation, for instance, to help doctors by simulating part of someone's body on the screen as in the examples below.

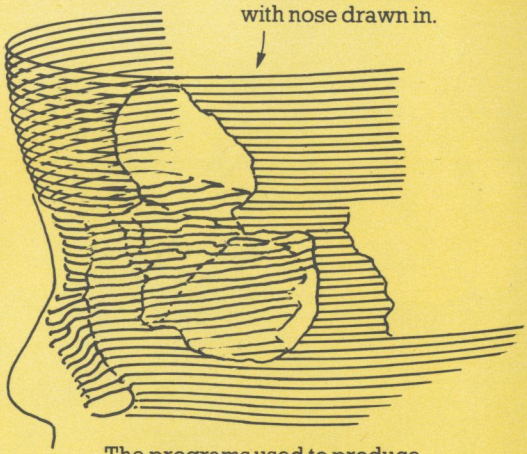
Computer-aided surgery

Computer graphics help plastic surgeons decide how best to operate on a patient before they even set foot in the operating theatre.

Computer image of damaged skull.

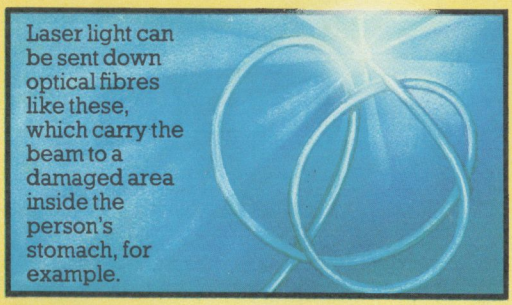


Side view of skull with nose drawn in.



The programs used to produce these images were developed from 3-D computer-aided design programs used in the design of aircraft at the McDonnell Douglas Corporation in the United States.

Since the early 1970s, computed tomography (CT) scanners have been used to produce X-ray cross-sections of a patient. Nowadays, computer graphics programs can stack a series of these cross-sections taken at, say, 2mm intervals, to make a complete 3-D image of the patient's bone structure. This image can be rotated on the screen so the surgeon can see all round it and even look at it from inside. The surgeon can experiment with different ways of operating on the patient by feeding different sets of data into the computer.



Laser light can be sent down optical fibres like these, which carry the beam to a damaged area inside the person's stomach, for example.

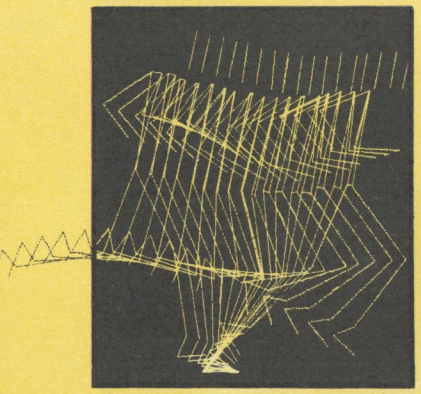
In the past, surgeons using lasers to perform delicate operations on damaged tissue had to rely on experience and guesswork to judge how powerful a beam to use. This sometimes resulted in failed operations.

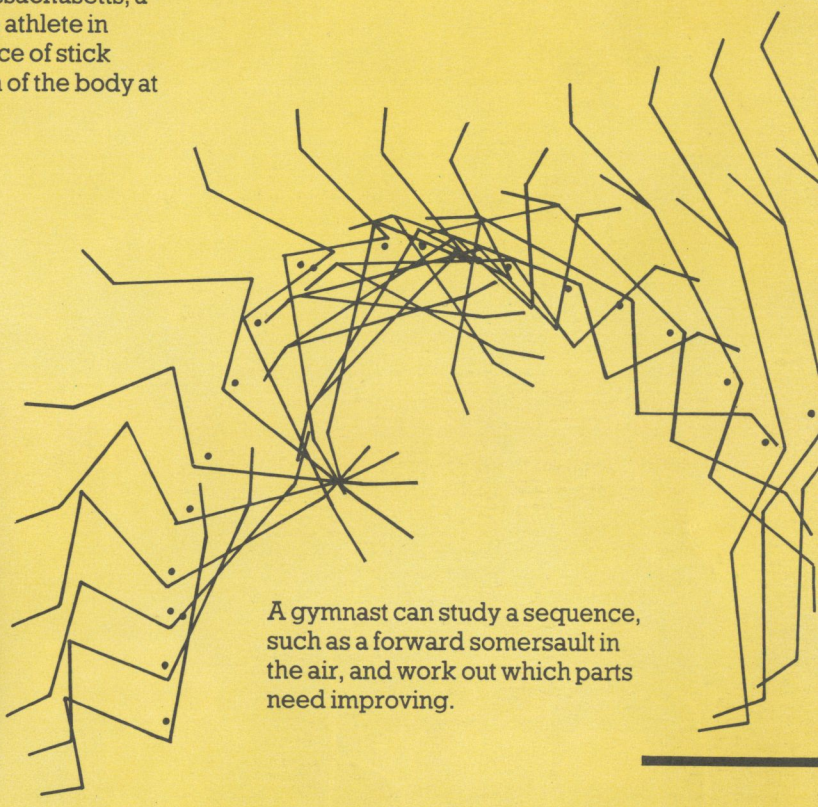
Nowadays, computers can help them by displaying a heat picture of the affected area. The surgeon can adjust the power and direction of a simulated laser beam until it is just right and the surgeon knows how much to use in the operation.

Computers and gold medals

Computers can help athletes make more efficient use of their bodies and perfect their movements. Using a technique developed at Computerized Biomechanical Analysis Incorporated in Massachusetts, a computer analyses film of the athlete in motion. It produces a sequence of stick pictures showing the position of the body at split second intervals.

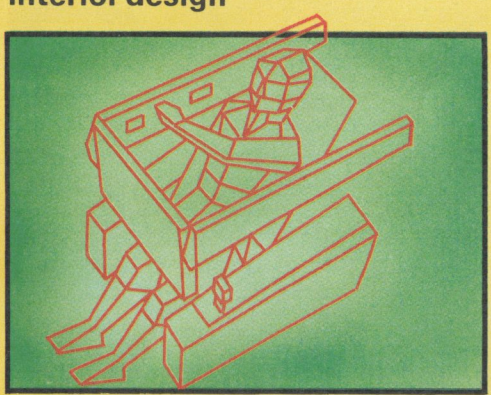
The computer can work out whether a runner would go faster if he or she leant forwards more or took longer strides.





A gymnast can study a sequence, such as a forward somersault in the air, and work out which parts need improving.

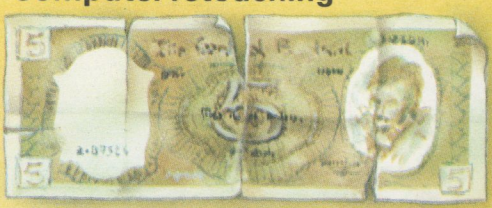
Interior design



Computer graphics can help design places where people work or sit, such as aeroplane or car interiors, so that they are convenient and comfortable. A 3-D graphics model that moves like a human is placed in the design on the screen and the design adjusted so that there is maximum leg room and any controls are in reach.

*See page 22.

Computer retouching



A graphics computer with an electronic retouching program can help make colour transparencies of damaged pictures which show the pictures looking as good as new. For instance, the computer could make a picture of an old and worn banknote showing it in mint condition. An image of the original is fed into the computer using a digitizing camera* and a copy appears on the screen. A scratch or tear can be made invisible by using a light pen to copy pixels from a next-door area and scattering them over the tear. A new colour transparency can then be made from the picture and used for reproduction in books, for example.

Computers and special effects

Some of the most exciting special effects in films and TV are computer-generated. Once an image is in the computer's memory it can be manipulated in different ways, some of which are shown on this page. Microcomputers do not usually have enough memory to store graphics with the same resolution and number of colours as film or TV, so larger minicomputers are used. For very sophisticated graphics, such as the computer-generated scenes in Walt Disney's film *Tron*, even more powerful mainframe computers are used. These can carry out 20 million calculations per second which is about 60 thousand times faster than most microcomputers. It can still take up to several minutes to create a single frame, or still picture, for the film.

Things the computer can do to a picture

It does not cost less to use computers to produce special effects but they can do things that were previously impossible or time-consuming. Objects can be transformed into other objects, pass right through each other, or change from being opaque to transparent. The camera can appear to zoom in from miles away and then fly round an object at a distance of only a few inches.

The picture can be squashed or stretched by changing the scale of the X and Y axes.

The picture can be scaled down and moved around the screen, leaving a trail of copies behind.

The X and Y co-ordinates can be mapped on to a shape, such as a wavy shape. Movement can be added so that it looks like a flag blowing in the wind.

The computer can change the colours or resolution of the picture for a different effect.

Digitizing a picture

Digitizing means converting information into numbers so the computer can work with it. Pictures, photographs or film can be fed into the computer using a digitizing camera. This divides the picture up into tiny areas and gives each area a number representing colour and brightness. The numbers are stored in part of the computer's memory called the digital frame store. This is similar to a home computer's graphics page* but it can store pictures of a higher resolution and more colours. The computer follows the numbers to reproduce the picture on the screen. It creates different effects by working on these numbers and the co-ordinates of the picture.

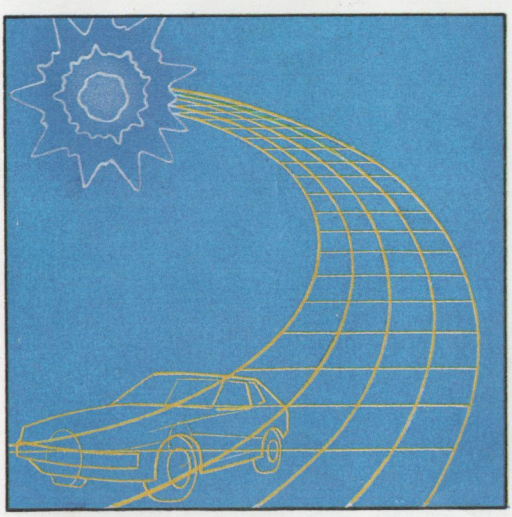
*The graphics page is explained on page 4.

Motion control

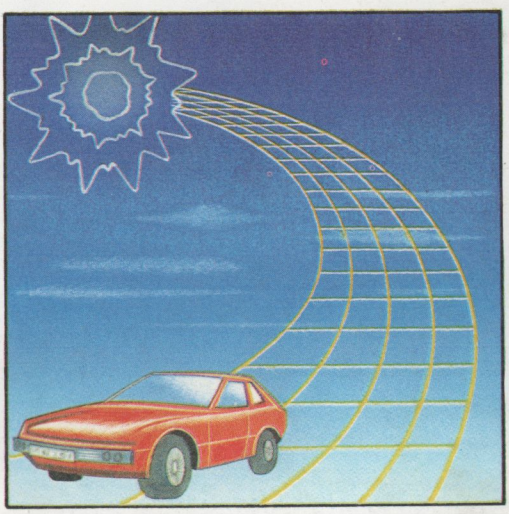
Motion control allows a film director to control the movement, positioning and focus of the camera with a computer. This gives complete accuracy to any shot. The director can preview the film sequence in a computer simulation on the screen and experiment with different ideas. Motion control is especially useful for films where several different shots need to be superimposed, for instance, real actors against a computer-generated background, as the computer can make sure everything will fit together perfectly.

© MCMLXXXII Walt Disney Productions

These actors in the film *Tron* were filmed against a plain colour background which was then filtered out using a photographic process and replaced with the computer-generated one. The process is called "chromakey" in TV production and "matting" in film. As well as in *Tron*, it was used in the *Star Wars* series.



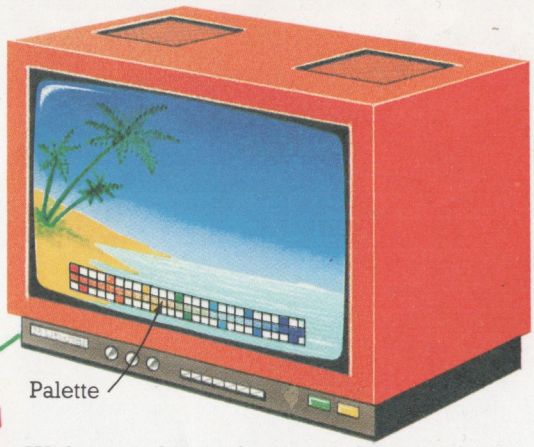
A director can simulate a commercial on a computer screen and show it to the client for approval before spending money on filming. This picture shows a simulated "animatic" of a commercial.

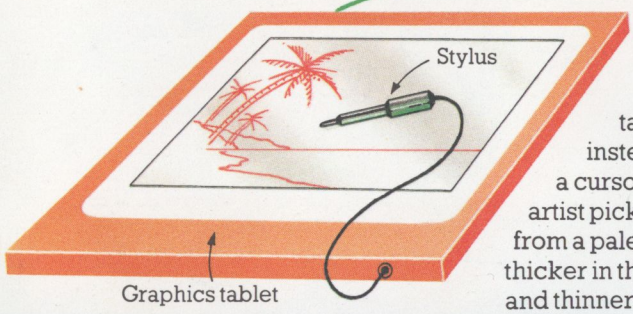


For the final version of the commercial, the car, the half-computerized and half-real background and any text are all shot under motion control so they fit together.

TV graphics

Film and television graphics, such as titles, credits, maps and charts, can be created on a computer, combining letters, pictures, colour, animation and even the special effects described on the previous page. Artists use special computer graphics systems and programs and work directly on high resolution screens.





With a graphics software package such as *Flair* or *Paint Box*, the artist uses a graphics tablet instead of a drawing board and a stylus instead of a brush. As the artist draws on the pad, a cursor moves round the screen leaving a line. The artist picks colours and can even mix different shades from a palette displayed on the screen. Colours appear thicker in the picture if the stylus is pressed down firmly and thinner where pressure is light.

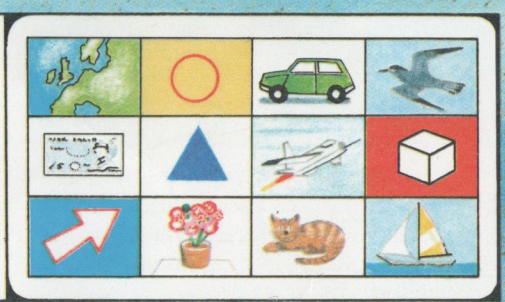
The program automatically draws straight lines and circles. Text can be added to the picture.

An existing TV picture can be fed into the computer and an artist can change it and add text to it, or even add parts of other pictures.

The computer can simulate the appearance of oils, water-colours, chalks, pastels or airbrushing and the artist can choose different sizes of brushstroke. Colours in the picture can be changed or parts of it erased or painted over. When it is finished, it can be stored on disk like any other computer data.

Computer library

The computer can store a "library" of pictures, parts of pictures or symbols. The artist can view the pictures in the library a group at a time or choose one from an index. The picture can be blown up to any size and worked on or incorporated into another picture.



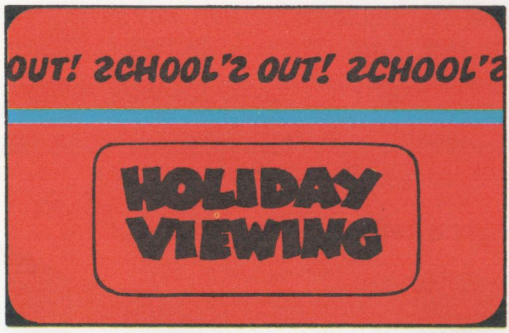
Caption generators

Typefaces available in any size

This is an example of Times
ABCDEFGabcdefg

This is an example of Baskerville
ABCDEFGabcdefg

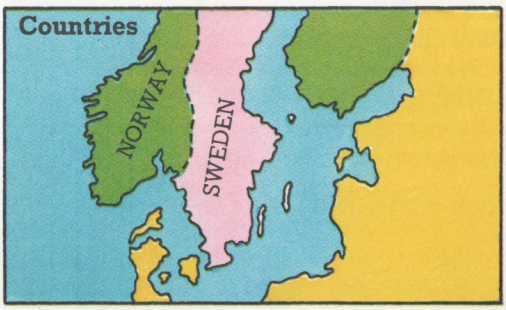
This is an example of Helvetica
ABCDEFGabcdefg

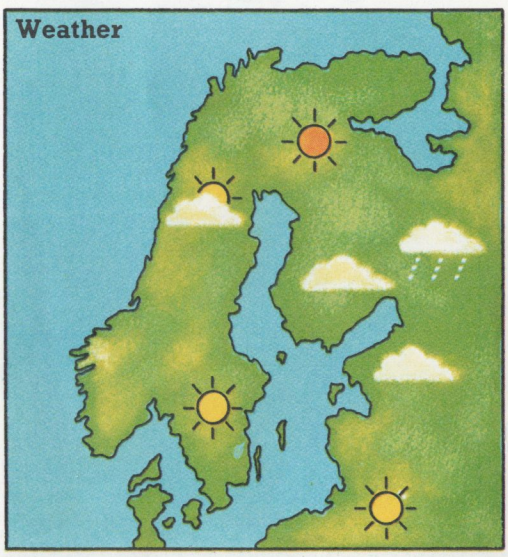


Film and TV programme credits and captions can be produced by a special computer called a caption generator. It stores different "fonts", or styles, of letters in different colours. The letters are smoother and of better quality than those produced on a home computer and the text can be scrolled sideways, called crawl, or up and down (roll) very evenly, unlike the jerky line-by-line scrolling on a home computer.

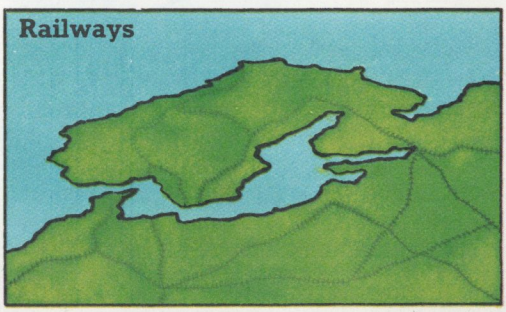
Maps and charts

Standard formats and designs, such as maps and charts used regularly on news and current affairs programmes, can be stored on disk. They can be recalled when needed, made larger or smaller, coloured, or rotated to give a different perspective.

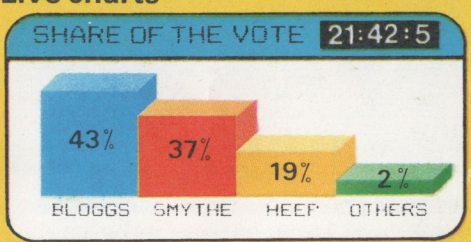




Standard shapes, such as the map above, can have overlays with different features or text added to them to show different information. These overlays are also stored on disk and positioned on the basic design when needed.



Live charts

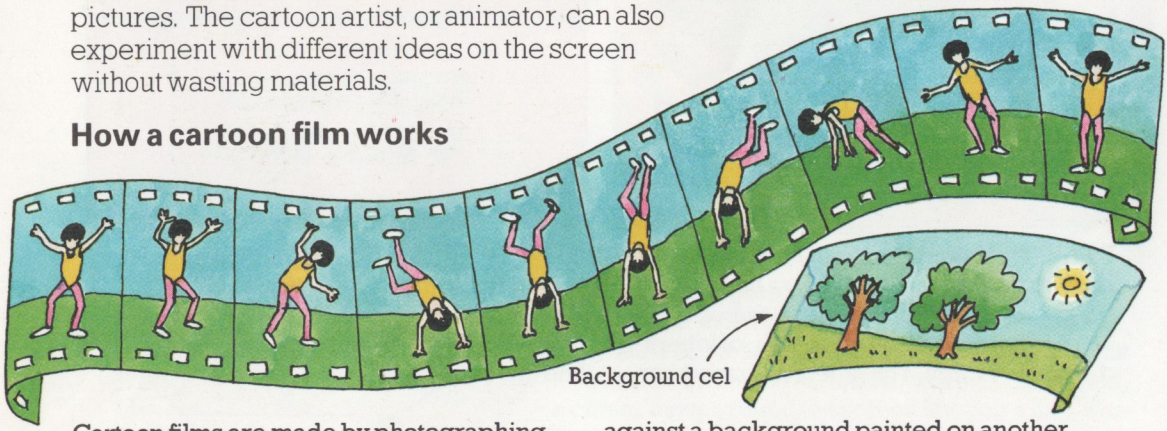


Charts and graphs can also be created very quickly and updated live during a broadcast, for example, to show election results coming in.

Animated graphics

These pages show another way in which computer graphics are used in films and film-making to save time and effort. Cartoons used to be completely hand-drawn. Now, using a process called in-betweening, computers can create many of the pictures. The cartoon artist, or animator, can also experiment with different ideas on the screen without wasting materials.

How a cartoon film works

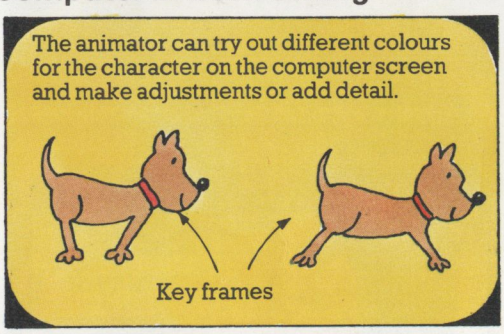


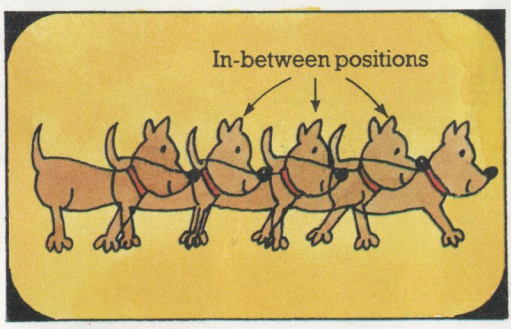
Cartoon films are made by photographing lots of different frames, or still pictures, each with the character drawn in a slightly different position. Each character position is drawn on to a separate cel, or piece of transparent film, which is photographed against a background painted on another cel. When the film is run at speed, it looks as if the character is moving. Film is usually projected at 24 frames per second, so 1,440 different pictures need to be drawn for one minute of film.

How cartoons used to be made

Cartoon films used to be drawn by teams of animators. The animation director would draw the key frames (indicated in red) showing the main positions of the character. These would show, for example, the furthest points to which the character's foot, knee and arm move when walking. Then, animators would draw in all the "in-between" positions.

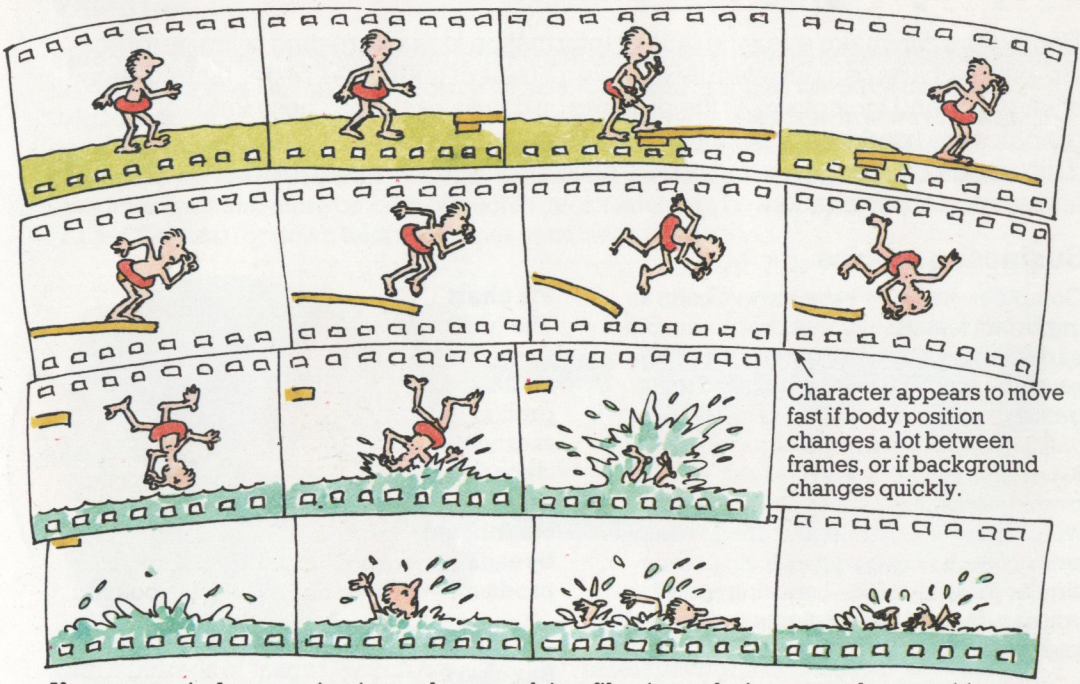
Computer in-betweening


The animator can try out different colours for the character on the computer screen and make adjustments or add detail.
Key frames



Nowadays, computers cut out the need to draw all the in-between pictures by hand. The animator draws the key frames on a graphics tablet or on the screen with a light pen. The computer looks at two consecutive frames and works out each in-between position by dividing the distances between corresponding points into fractions. This gives it new sets of points which it joins up to make the in-between shapes. The frames are then photographed separately to make cels. There is a program on page 36 which demonstrates the principles of in-betweening.

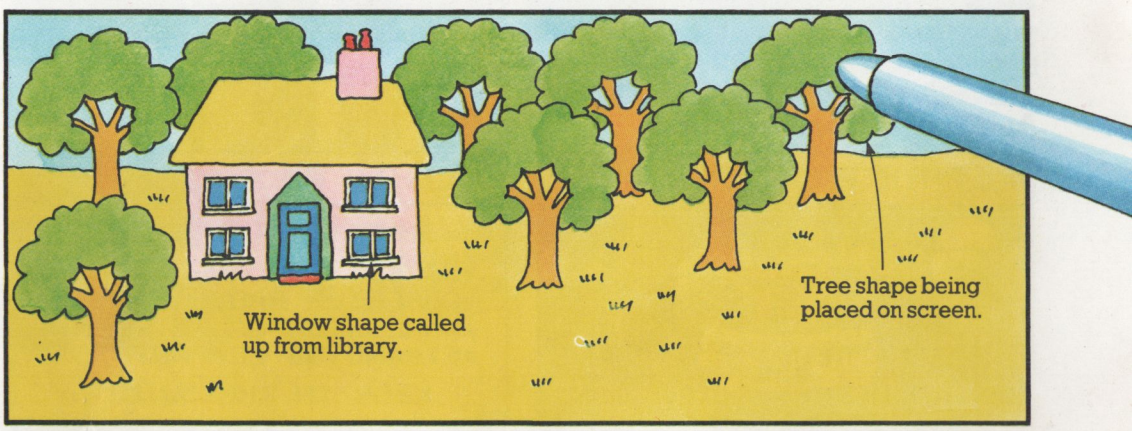
Character appears to move slowly if body position and background do not change much between frames.



Character appears to move fast if body position changes a lot between frames, or if background changes quickly.

If computer in-betweening is used too much in a film, it results in a smooth, unexciting, lifeless flow of movement from one position to the next. The animator can prevent this by drawing more key frames for some movements than others, for example, the movements that give a character personality, such as a funny walk. The animator can also choose how many in-between frames are needed between each key frame. The more distance there is between the position of the character in each frame, the faster the movement appears.

Painting the backgrounds



The backgrounds for the animation are also drawn on a colour computer with a special painting and drawing program. The artist can build up a library of shapes to use in the film and even incorporate photographs from the real world digitized by a camera as described on page 22.

The background artist can choose a shape or symbol and then leave copies of it anywhere on the screen. For instance, by calling up a tree shape from the library the artist can create a whole forest just by touching the pen on the screen or pad wherever a tree is needed.

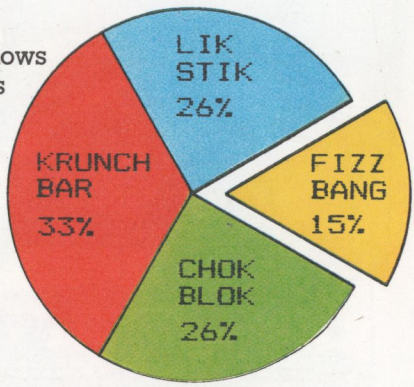
Displaying information

Computers can make even the dullest information look interesting using graphics. Graphs or charts can put over the meaning of sets of figures far better than columns of numbers and look nicer, as the pictures on this page show. These kinds of graphics are used a lot in business. Huge amounts of information can be kept on disk outside the computer and once a disk is in use it only takes a few seconds to find a piece of information.

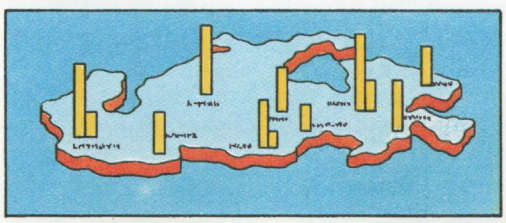
Business graphics

Computer graphics have always been an important feature of home computers but early business microcomputers had little or no graphics. Then people realized that better graphics would be useful for displaying information such as sales reports, projections and budget breakdowns. Now, there are programs which generate graphs and charts when sets of figures are typed in. You can buy similar programs for a home computer so you can do your own budgets and record-keeping.

Pie chart
The "pie" shows a company's total annual profits. The segments show the percentage contributed by each product.

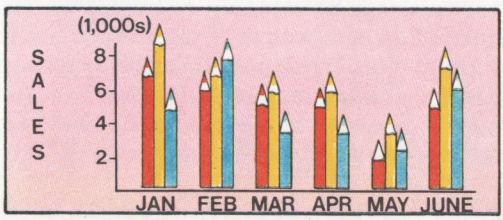


3-D chart



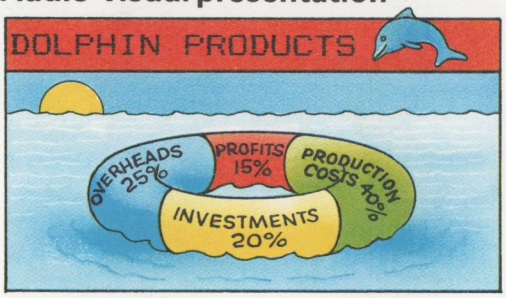
This chart shows how many sales are made in each area of a country.

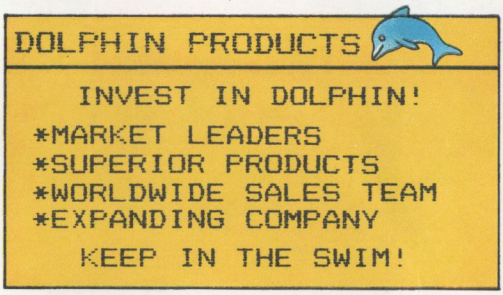
Bar chart



This bar chart uses coloured pencils instead of bars to show how well each colour sells each month.

Audio-visual presentation

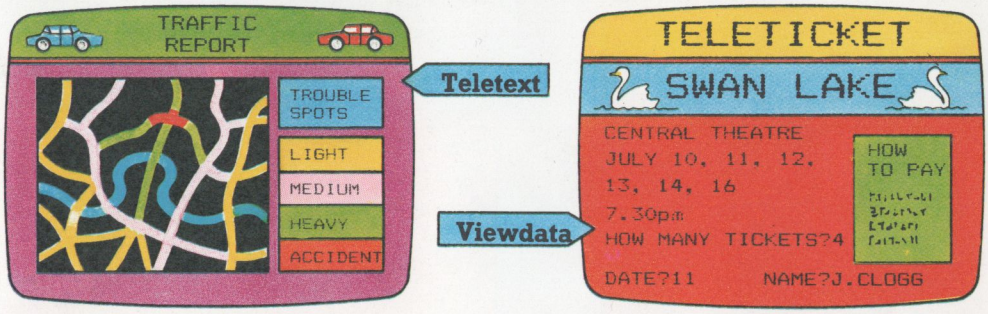




Computers are used in the production of colour transparencies for audio-visual presentation. This is a system of presenting information visually, with a spoken or taped commentary. It is used a great deal in business, for example, to tell prospective clients about a company or to give a report to shareholders. Originally, graphs and charts would be worked out, drawn up and assembled with text by an artist and then photographed. Now, the whole display can be designed on a computer using a program to generate the charts which also allows the artist to design a colour background and add text. The display is then photographed from the screen using a camera like the one shown on page 11.

Videotex

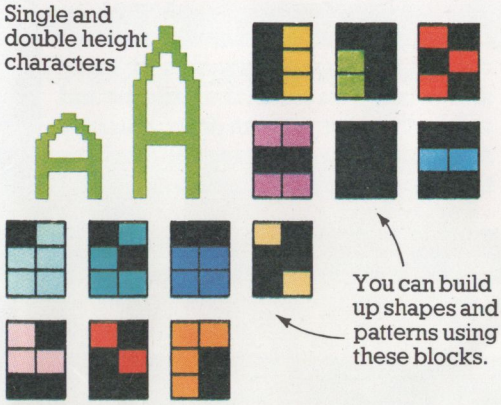
Videotex is a system of transmitting information from central computerized databanks for display in people's homes on ordinary TV sets. Colourful graphics are often used to brighten up the screens, or "pages" of information. This was one of the first ways in which computer graphics were shown on TV screens and originally they had a distinctive, chunky look. Newer Videotex systems, such as the Canadian Telidon, can make much smoother lines and more colours. Videotex information can be sent by broadcast signals (called Teletext) or down telephone lines or cable (Viewdata).

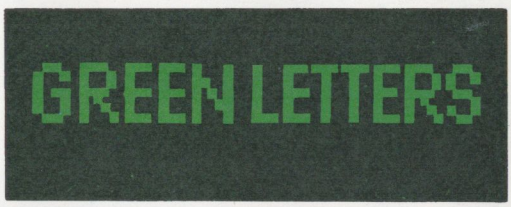


Teletext is a kind of electronic newspaper with hundreds of "pages" which are continually being updated. You choose a page to look at by pressing numbers on a keypad. The pages are broadcast as part of the TV programme signals.

Viewdata information is sent down cables or telephone lines to a small terminal or computer with an adaptor and shown on a TV screen. Viewdata is interactive, which means the user can also send messages back to the main computer.

Teletext graphics





Control code for text colour

```
PRINT CHR$(130);
"GREEN LETTERS"
```

Control codes tell the computer which colours to make the letters and background.

The original, chunky style of lettering and graphics on Viewdata and Teletext screens is known as teletext style. This kind of screen display only uses 1K of memory but provides eight colours, two sizes of text characters and a flashing effect. Graphics are built up from pre-defined graphics blocks made up of six pixels each. This saves on memory, as the computer does not have to control each pixel individually.

Some home computers, such as the BBC, have a separate mode for making teletext characters and graphics. This is mode 7 on the BBC. You make the computer print in colour, or make double-height or flashing characters, by putting control codes in the PRINT lines as shown above. When the program is run, the control code appears as a space in the text and it affects everything that follows it until another code is printed.

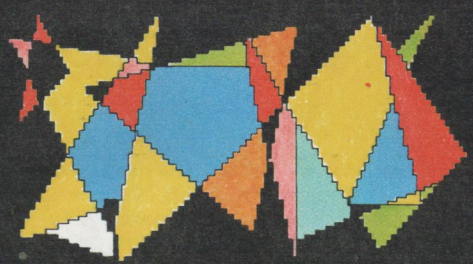
Computer art

In the early days of computers (the late 1950s and 1960s), the only people who could use them were highly-trained computer technicians who understood complex programming languages. Some big companies, though, such as IBM, wanted to persuade people that computers were interesting and useful to people other than scientists, so they commissioned artists to work with their technicians and explore computer art. Programming languages that were easier to use and more suitable for graphics were developed and special graphics equipment such as light pens and graphics tablets were invented. These pages show some exciting examples of recent computer art.

Why artists use computers

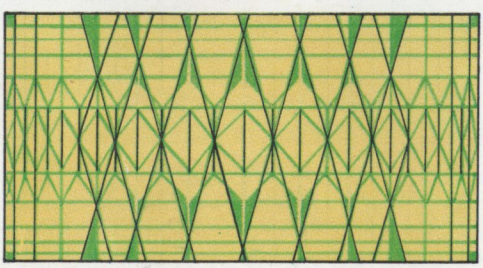
A computer is a different kind of artist's tool from a paintbrush or a crayon. It can generate perfect geometric shapes and repeat them in different positions or sizes to make abstract patterns. It can introduce a random element into a picture by picking a random number for an angle, say, or a random colour. Here, a basic shape is drawn by the artist on a computer which then works on the shape and repeats it to make the whole pattern.

Home computer art

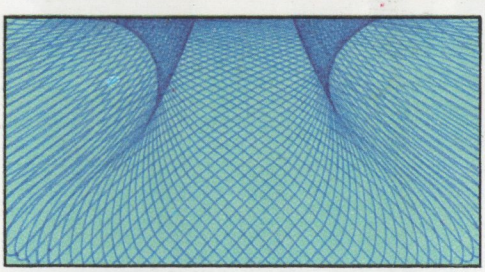


You can buy programs for a home computer which generate different shapes, for example, squares, spheres and cones. You can make patterns and colour them, or you can draw your own shapes and the computer will work on them according to its program.

Weaving pictures



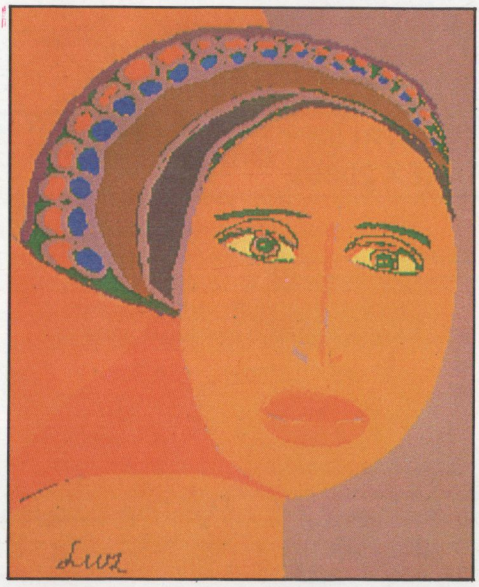
These patterns are based on weaving designs, using a program called *Metaloom* which runs on a minicomputer. The program can "weave" up to 5,000 ×

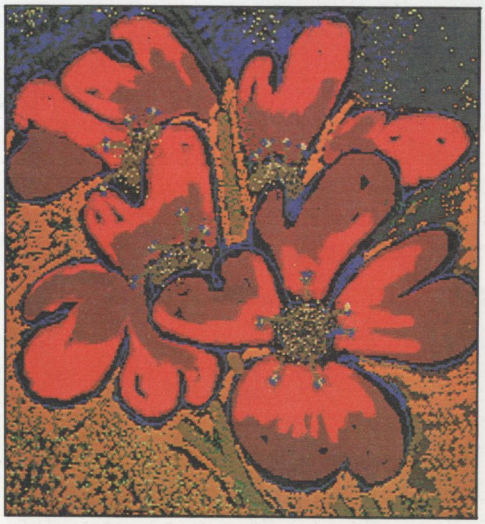


5,000 "threads" and can make a 2,000 × 2,000 thread design in three seconds. It can also produce weaves that look as if they are being pulled or stretched.

Painting with a computer

These pictures were created by an artist called Luz Bueno. She draws shapes on a graphics tablet or works with images from a digitizing camera (see page 22). She can mix over 4,000 colours on her computer and works on a monitor screen with a resolution of 754 × 482 pixels.

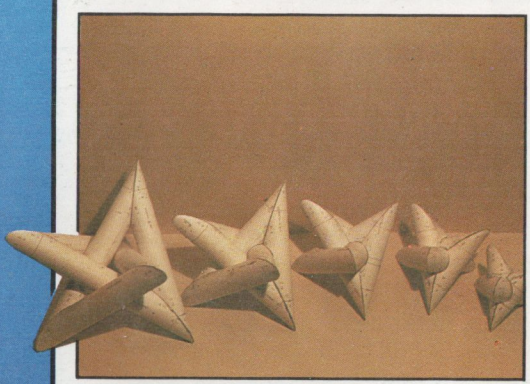




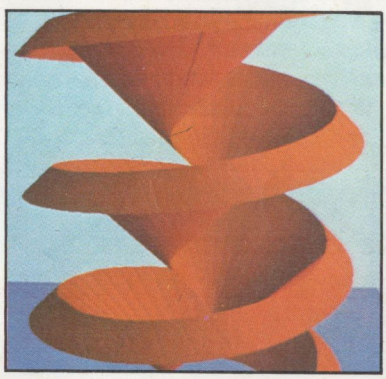
Computer art on paper

Computer art can be photographed from the screen using a special camera, or printed out on a high resolution printer using very small dots.* However, the colours are not usually as rich on paper as those seen on a computer screen.

Computer-assisted sculpture



Frank Smullin uses a computer to help work out the designs for his sculptures. Calculating where to cut these cylinders so they will all fit together perfectly would be extremely difficult without a computer.



Ronald Resch makes sculptures out of flat sheets of metal bent along scored lines. He simulates a 3-D model of the sculpture on a computer which works out a diagram of the unfolded sheet of metal showing where to score lines. The computer can then be attached to a machine that scores the metal.

*See pages 10-11 for pictures of camera and printer.

Graphics programs to run

In the following section of the book there are four graphics programs. The programs are specially written so that with minor alterations they will run on the BBC, Spectrum (Timex 2000), Apple and Dragon computers. To convert the programs to run on one of these computers, you need to refer to the Conversion Chart on pages 44 and 45. This lists the different commands for each of the four computers. On these two pages there are instructions for how to use the Conversion Chart and a few other points to watch out for. You will be reminded alongside the program listing itself whenever you need to make a conversion, but it is helpful to read these two pages first.

Using the Conversion Chart

Part of Conversion Chart

Pseudo-commands	Description of command	BBC	SPECTRUM (TIMEX 2000)	DRAGON	APPLE
gcolour N	Set the colour for drawing lines.	GCOL 0,N	INK N	COLOR N,1	HCOLOR=N

Pseudo-commands listed in Conversion Chart.

Description of what each command does.

Commands for different computers to be substituted for pseudo-commands.

Program line marked with an asterisk.

```
✸430 gcolour C%(J%)
```

Pseudo-command in small letters.

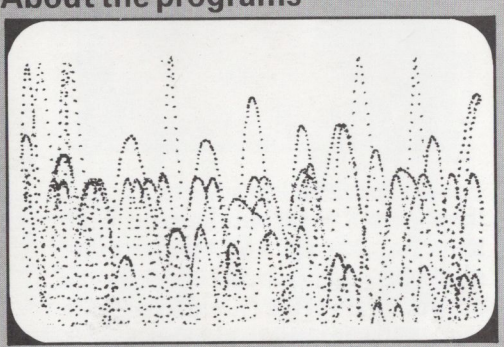
```
430 GCOLO,C%(J%)
```

Program line converted for the BBC.

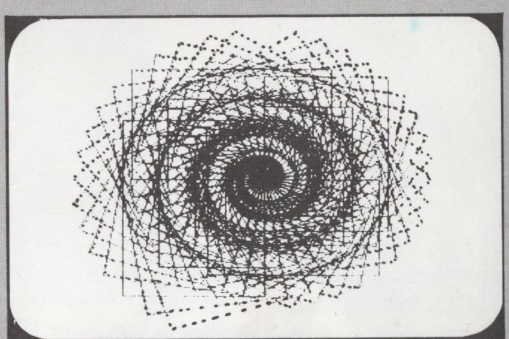
If you have a different computer to the four in the Conversion Chart, you may still be able to convert the programs to run on your computer. The chart has a brief description of what each command does, so you can try looking in your manual to find the corresponding command for your computer.

Different computers understand different commands, so some of the program lines in the listings contain "pseudo-commands" instead of the command for any particular computer. These lines are marked with an asterisk (✸) to tell you to refer to the Conversion Chart and the pseudo-commands are printed in small letters. The Conversion Chart lists the pseudo-commands and gives you the correct command to substitute for your computer. There is an example above showing how to use the Conversion Chart.

About the programs



All the programs use two or more colours. The Pattern Generator and Turtle Graphics

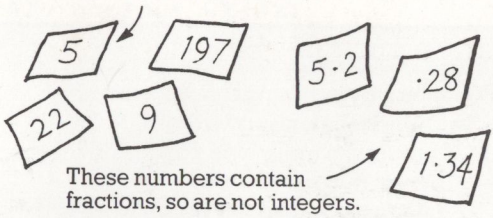


programs let you create your own patterns using two different techniques.

Other changes you may need to make

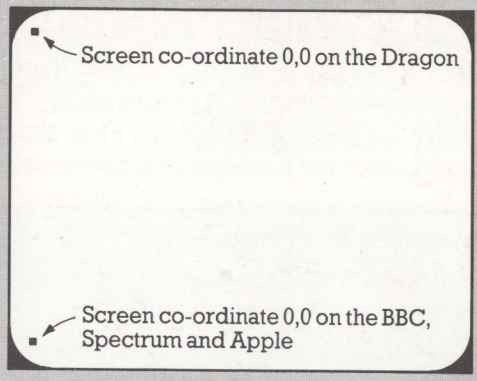
1 In some of the programs, whole lines need changing or extra lines adding for a particular computer. These are given at the end of each program listing.

2 These numbers are integers (whole numbers).



If you have a Spectrum (Timex 2000) or a Dragon, leave out all the % signs in the program listings. These symbols indicate a special kind of variable called an integer variable. Using them makes the program run faster. An ordinary variable, such as W or CL, is a label for an area in the computer's memory which can store a number. An integer variable only stores a whole number (integer) and not a fraction. The BBC and Apple computers understand the % sign to mean integer variable, but not the Spectrum or Dragon. The programs will work in the same way without them, but slightly slower.

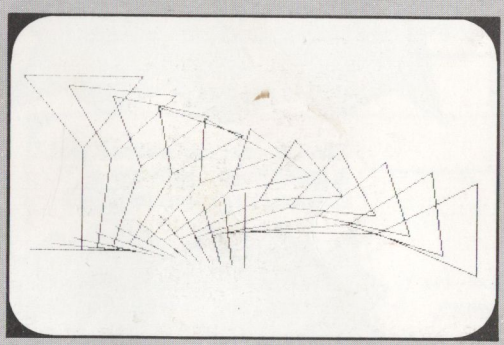
Special note for Dragon users



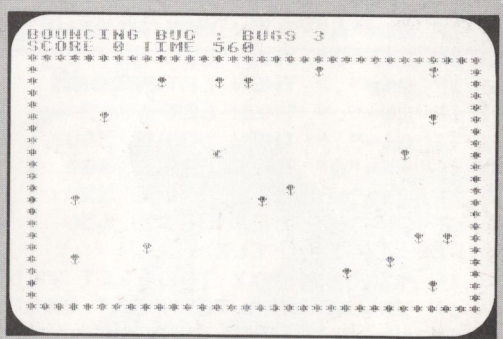
In all the programs, you move a point around the screen to draw a shape, or to move the bug in the Bouncing Bug game, by pressing four different keys for up, down, left and right. For the Dragon you need to change over the keys which move the point up or down. (You will be told where to do this in the notes beside the program listings.) This is because on the Dragon, the screen co-ordinate 0,0 is in the top left corner of the screen instead of the bottom left, as in the other three computers.

Using colour in the programs

Computers use numbers to represent colours. When you run the programs, you are asked to type in the numbers which represent the colours you want to use. These numbers vary on different computers, for example, 1=red on the BBC but 1=blue on the Spectrum. Look in your manual to find which numbers to type in for which colours on your computer.



The In-betweening program shows how the computer changes one shape into another.



The last program, Bouncing Bug, is an animated game.

Pattern generator

This program draws colourful, constantly changing patterns on the screen. You can choose whether you want the patterns to be drawn as trails of different coloured dots or whether you want the dots joined up to make different coloured shapes (see bottom of opposite page). You can alter the speed and direction of the dots by pressing certain keys. First, though, try the program without pressing any keys to see what kind of patterns the computer produces by itself.

Running the program

When you run the program, the following instructions appear on the screen:

TYPE 1 FOR A SHAPE
TYPE 2 FOR DOTS

If you type 1, lines will join the dots up to make triangular shapes. If you type 2, the dots are not joined up. You are then asked to type in the numbers for the colours you want the dots to be (see previous page). Type the numbers in figures, not words. The dots will then appear on the screen and begin to move. You can affect their movement by pressing these keys:

"A" bends the trail upwards.
"Z" bends the trail downwards.
"," bends the trail to the left.
"." bends the trail to the right.
"Q" changes the direction of the dots from left to right (or vice versa).
"W" changes the direction of the dots from up to down (or vice versa).
"S" freezes the pattern on the screen.
Pressing the space bar wipes the pattern off the screen and begins a new pattern.

```
 10 LET CL%=3
*20 LET WI%=screen width
*30 LET HI%=screen height
 40 LET CX%=WI%/2
 50 LET CY%=HI%/2
 60 LET MX%=WI%/20
 70 LET MY%=HI%/20
 80 DIM X%(CL%):DIM Y%(CL%):DIM V%(CL%):
 DIM W%(CL%):DIM C%(CL%)
 90 GOSUB 680
100 PRINT "TYPE 1 FOR A SHAPE"
110 PRINT "TYPE 2 FOR DOTS"
120 INPUT D%
130 IF D%<1 OR D%>2 THEN GOTO 120
*140 gmode
150 GOSUB 560
160 LET V%=0
170 LET W%=0
*180 LET Q$=inkey$
190 IF Q$="A" THEN LET W%=2
200 IF Q$="Z" THEN LET W%=-2
210 IF Q$="," THEN LET V%=-2
220 IF Q$="." THEN LET V%=2
230 IF Q$=" " THEN GOSUB 560
240 IF Q$="Q" THEN GOSUB 480
250 IF Q$="W" THEN GOSUB 520
260 IF Q$="S" THEN GOSUB 650
270 FOR J%=1 TO CL%
280 IF V%(J%)<-MX% THEN LET V%(J%)=-MX%
290 IF V%(J%)>MX% THEN LET V%(J%)=MX%
300 IF W%(J%)<-MY% THEN LET W%(J%)=-MY%
310 IF W%(J%)>MY% THEN LET W%(J%)=MY%
320 LET V%(J%)=V%(J%)+V%
330 LET W%(J%)=W%(J%)+W%
```

Number of trails of dots. You can change this number if you want more trails.

NB If you have a Spectrum (Timex 2000), type in 254 for the screen width and 174 for the screen height in this program instead of the numbers printed in the conversion chart. There are some extra conversions for the Spectrum (Timex 2000) at the end of the program.

Program lines marked with an asterisk contain a pseudo-command which needs converting for different computers. Find the correct command for your computer in the Conversion Chart on pages 44 and 45.

Change over the letters A and Z in these lines if you have a Dragon (see previous page).

If you have an Apple, replace J% with J and I% with I from line 270 until the end of the program. Leave out all the % signs if you have a Spectrum or a Dragon (see previous page).

```
340 LET X%(J%)=X%(J%)+V%(J%)
350 LET Y%(J%)=Y%(J%)+W%(J%)
360 IF X%(J%)<1 THEN LET X%(J%)=1:LET V%(J%)=-V%(J%)
370 IF Y%(J%)<1 THEN LET Y%(J%)=1:LET W%(J%)=-W%(J%)
380 IF X%(J%)>WI% THEN LET X%(J%)=WI%:LET V%(J%)=-V%(J%)
390 IF Y%(J%)>HI% THEN LET Y%(J%)=HI%:LET W%(J%)=-W%(J%)
400 NEXT J%
410 move X%(CL%),Y%(CL%)
420 FOR J%=1 TO CL%
430 gcolour C%(J%)
440 IF D%=1 THEN draw X%(J%),Y%(J%)
450 IF D%=2 THEN dot X%(J%),Y%(J%)
460 NEXT J%
470 GOTO 160
480 FOR I%=1 TO CL%
490 LET V%(I%)=-V%(I%)
500 NEXT I%
510 RETURN
520 FOR I%=1 TO CL%
530 LET W%(I%)=-W%(I%)
540 NEXT I%
550 RETURN
560 clg
570 FOR I%=1 TO CL%
580 LET X%(I%)=CX%
590 LET Y%(I%)=CY%
600 LET V%(I%)=rnd(1)*12-6
610 LET W%(I%)=rnd(1)*12-6
620 NEXT I%
630 move CX%,CY%
640 RETURN
650 LET Q$=inkey$
660 IF Q$<>" " THEN GOTO 650
670 RETURN
680 cls
690 FOR I%=1 TO CL%
700 PRINT "TYPE COLOUR
NUMBER FOR POINT ";I%
710 INPUT C%(I%)
720 NEXT I%
730 RETURN
```

These lines make the dots bounce off the edge of the screen.

Computer either joins the dots up (line 440) or not (line 450), according to whether you pressed 1 or 2 in lines 100-110.

Computer goes to these lines if you press "Q" and changes the direction of the dots from left to right or vice versa.

Computer goes to these lines if you press "W" to change direction of the dots from up to down or vice versa.

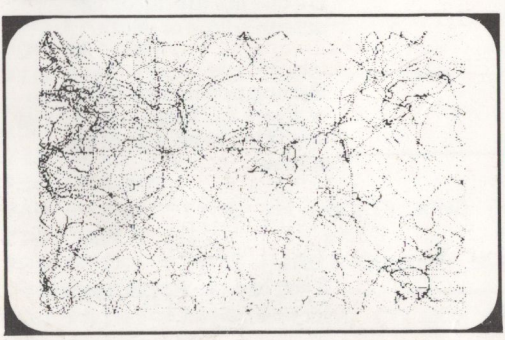
If you press "S", computer comes here. It waits until you press the space bar and then clears the screen and starts a new pattern.

Extra conversions for the Spectrum (Timex 2000)

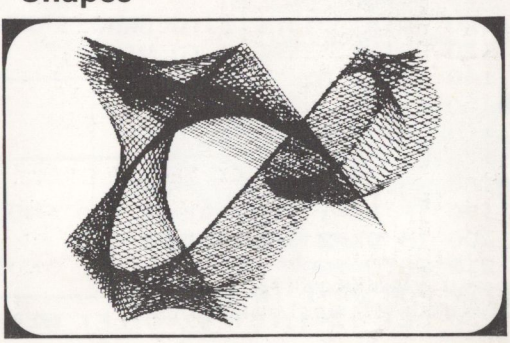
Include these lines in the program:

```
210 IF Q$="N" THEN LET V=-2
220 IF Q$="M" THEN LET V=2
415 LET OX=X(CL):LET OY=Y(CL)
440 IF D=1 THEN DRAW X(J)-OX,Y(J)-OY
455 LET OX=X(J):LET OY=Y(J)
```

Dots



Shapes



In-betweening

You can watch how the computer changes one shape into another with this program. It is a demonstration of simple in-betweening, which is described on pages 26 and 27. You draw two shapes on the screen and tell the computer how many in-between stages you want. You can get different effects by drawing your shapes at either side of the screen, or by drawing one very small shape inside a large one. The picture in the yellow box on page 3 was generated using this program.

Running the program

When you run the program, the computer asks you to type in a number for your background colour. Type in the number for your standard screen colour.* The computer then asks you for two more colour numbers. These are for your shapes and the in-between shapes.

The following instruction then appears:

TYPE 1 TO DRAW
TYPE 2 TO VIEW

Press 1 to draw your shapes. A tiny, flickering dot appears on the screen. You can move it around using the "A" and "Z" keys for up and down and the "," and "." keys for left and right.

Position the dot where you want to start your first shape and press the space bar. This fixes the point in the computer's memory. Move the dot to the next position and press the space bar. A line will join the two points up. Do this for all the points in your shape. As it stands, the program allows you five points for each shape but you can increase the number in line 20 for more complicated shapes.

While you are drawing, the computer tells you whether you are on shape 1 or shape 2 and how many points you have left.

When you have drawn both shapes, the computer asks you again if you want to DRAW or VIEW. This time, press 2. It then asks "HOW MANY IN-BETWEEN SHAPES?". Type in the number of in-between shapes you want the computer to draw. Try about 10 to start with. Your shapes will then appear on the screen in one colour and the computer will draw the in-between shapes in the other colour you specified.

```
*10 gmode MODE 2
 20 LET L%=5
*30 LET WI%=screen width 1279
*40 LET HI%=screen height 1023
 50 LET CX%=WI%/2
 60 LET CY%=HI%/2
 70 LET MX%=WI%/100
 80 LET MY%=HI%/100
 90 DIM X%(L%,2):DIM Y%(L%,2)
*100 cls
 110 GOSUB 760
 120 PRINT "TYPE 1 TO DRAW"
 130 PRINT "TYPE 2 TO VIEW"
 140 INPUT D%
*150 clg
 160 LET X%=0
 170 LET Y%=CY%
*180 move X%,Y%
 190 IF D%=1 THEN GOSUB 220
 200 IF D%=2 THEN GOSUB 480
 210 GOTO 120
 220 FOR J%=1 TO 2
 230 LET I%=1:LET V%=0:LET W%=0
```

Leave out % signs if you are using a Dragon or a Spectrum (Timex 2000).

MX and MY are the horizontal and vertical distances the dot will move each time you press a direction key while drawing your two shapes. Each distance is a one hundredth of the screen height or width.

Starting position for dot at co-ordinate 0,CY which is half way up the left hand side of the screen.

If you have an Apple, replace J% with J in line 220.

*0 for the BBC, 7 for the Spectrum, 0 for the Apple and 1 for the Dragon.

```
✸240 clg
 250 GOSUB 690
✸260 LET Q$=inkey$
```

Change keys A and Z over in lines 270 and 280 if you are using a Dragon.

```
 270 IF Q$="A" THEN LET W%=MY%
 280 IF Q$="Z" THEN LET W%=-MY%
 290 IF Q$="," THEN LET V%=-MX%
 300 IF Q$="." THEN LET V%=MX%
✸310 gcolour C1
 320 LET X%=X%+V%
 330 LET Y%=Y%+W%
 340 IF X%<0 THEN LET X%=0
 350 IF Y%<0 THEN LET Y%=0
 360 IF X%>WI% THEN LET X%=WI%
 370 IF Y%>HI% THEN LET Y%=HI%
✸380 dot X%,Y%
 390 IF Q$=" " THEN GOSUB 640
✸400 gcolour BC
 410 GOSUB 730
✸420 dot X%,Y%
 430 LET V%=0
 440 LET W%=0
 450 IF I%<L%+1 THEN GOTO 260
 460 NEXT J%
 470 RETURN
 480 INPUT "HOW MANY
     IN-BETWEEN SHAPES?";F%
 490 FOR Z%=0 TO F%
 500 PRINT "FRAME NO ";Z%+1
 510 LET X%=(X%(1,1)*(F%-Z%)+X%(1,2)*Z%)/F%
 520 LET Y%=(Y%(1,1)*(F%-Z%)+Y%(1,2)*Z%)/F%
✸530 move X%,Y%
 540 FOR I%=1 TO L%
 550 LET X%=(X%(I%,1)*(F%-Z%)+X%(I%,2)*Z%)/F%
 560 LET Y%=(Y%(I%,1)*(F%-Z%)+Y%(I%,2)*Z%)/F%
✸570 gcolour C1
✸580 IF Z%<1 OR Z%>F%-1 THEN gcolour C2
✸590 draw X%,Y%
 600 NEXT I%
 610 GOSUB 710
 620 NEXT Z%
 630 RETURN
 640 LET X%(I%,J%)=X%
✸650 IF I%=1 THEN dot X%,Y%:LET X%=X%+MX%
 660 LET Y%(I%,J%)=Y%
✸670 IF I%>1 THEN draw X%(I%-1,J%),Y%(I%-1,J%)
 680 LET I%=I%+1
 690 PRINT "SHAPE ";J%;" : POINTS LEFT ";L%+1-I%
 700 RETURN
 710 FOR T%=1 TO 2000
 720 NEXT T%
 730 FOR T%=1 TO 100
 740 NEXT T%
 750 RETURN
 760 PRINT "TYPE IN BACKGROUND COLOUR"
 770 INPUT BC
 780 PRINT "TYPE FIRST COLOUR NUMBER"
 790 INPUT C1
 800 PRINT "TYPE SECOND COLOUR NUMBER"
 810 INPUT C2
 820 RETURN
```

Extra conversions

Apple:

```
105 VTAB(21)
485 HCOLOR=C2
580 IF Z>F%-2 THEN HCOLOR=C2
```

BBC:

```
 15 VDU 24,0;128;1279;1025;
 16 VDU 28,0,31,39,28
```

Spectrum (Timex 2000):

```
 40 LET HI=167
465 INK C1
500 PRINT AT 0,0;"FRAME
    NUMBER ";Z+1
535 LET OX=X:LET OY=Y
590 DRAW X-OX,Y-OY
595 LET OX=X:LET OY=Y
670 IF I>1 THEN DRAW X(I-1,J)
    -X,Y(I-1,J)-Y
690 PRINT AT 0,0;"SHAPE ";J;"
    : POINTS LEFT ";L+1-I
```

Dragon:

```
 10 PMODE 3
485 SCREEN 1,0
500 Delete this line
625 IF INKEY$<>" " THEN
    GOTO 625
692 GOSUB 710
694 SCREEN 1,0
```

This line asks you to choose how many in-between shapes you want the computer to draw. If you are using a BBC, put a comma instead of a semi-colon before F% at the end of the line.

If you have an Apple, replace J%, Z% and I% with J, Z and I in lines 460 to 620. Replace J% with J in lines 640 to 690.

If you press the space bar at line 390, computer comes here. If it is the first point in your shape, computer makes a dot. For other points it joins them up with lines.

If you have an Apple, replace T% with T in lines 710 to 740.

If you have a Dragon or a Spectrum, reduce the number 2000 in line 710 to make the program run faster. Try about 500.

Turtle graphics

Turtle graphics are created by moving a small pointer, or "turtle", around the screen. The turtle leaves a line behind it as it goes. The computer asks you for the length and colour of the line you want to draw and the angle through which you want to move the turtle. In the first part of this program, you build up a shape, line by line, in this way.

In the second part of the program, the computer works on your shape to create an intricate pattern according to further instructions from you. The questions you have to answer in both parts of the program are explained below. The pattern on page 1 of this book was created using this program.

Drawing the shape

When you run the program the computer prints this instruction on the screen:

TYPE 1 TO DRAW A SHAPE
TYPE 2 TO MAKE A PATTERN

First, type 1 to draw your shape. You can use 12 lines to make the shape and for each line the computer asks you the following questions:

ANGLE TO TURN? Type in a number between 0 and 360 telling the computer how many degrees you want it to turn the turtle through before drawing a line. The turtle always turns clockwise.*

LENGTH OF MOVE? Type in a number telling the computer how long to make the line. This depends on your screen dimensions so you need to experiment. Try 50 to start with and see how far the turtle moves. (If you finish your shape before using up all 12 moves, type 0 here and the computer will go on to the next stage of the program, which is drawing the pattern.)

DRAW (Y/N)? Type Y if you want the turtle to draw a line. Type N if you want it to move to a new position without drawing a line.

DRAWING COLOUR? Type your computer's code number for the colour you want the line to be drawn in. If you have a Dragon, press the space bar after each line has been drawn on the screen.

When you have used up all 12 moves, or typed 0 for LENGTH OF MOVE, the computer asks you whether you want DRAW or PATTERN again. This time, type 2 for PATTERN. (If you type 1, you will lose your shape and the computer will ask you the questions again for a new shape.)

Making the pattern

The computer can now use your shape to make a pattern. It asks you the following questions and then draws the shape over and over again, rotating and enlarging it a little each time according to your instructions.

STARTING SCALE? Type how large you want the first shape to be drawn. For instance, if you type 0.5, the first shape will appear half as big as you drew it. This means that you can draw your shape quite large and then reduce it.

NUMBER OF STEPS? Type how many times you want the shape to be repeated. Try about 30 to start with. The computer will only draw until a line reaches the edge of the screen, so you need to experiment with shapes that will fit on your screen when they are enlarged.

SCALE CHANGE? Type the amount you want the computer to enlarge the shape each time it rotates and draws it. Keep this number small, such as 0.05, or the pattern will reach the screen edge and the computer will stop drawing before it has completed many shapes.

ANGLE CHANGE? Type how many degrees you want the computer to rotate the shape each time it draws it. Try about 3 to start with.

When the computer has drawn the pattern, it asks you whether you want DRAW or PATTERN again. (If you have a Dragon, you need to press the space bar before it asks you this question.)If you type 1 for DRAW, you lose the pattern and can start again. If you type 2 for PATTERN, you can use the same shape but make a different pattern by typing in different numbers for the above questions.

*On a Dragon, the turtle always moves anti-clockwise.

Extra conversions

Include these lines if you have a BBC:

```
15 VDU 24,0;128;1279;1025;
16 VDU 28,0,31,39,28
```

Include these lines if you have a Dragon:

```
 10 PMODE 3
115 SCREEN 1,0
250 LET C=2
335 IF INKEY$="" THEN GOTO 335
385 SCREEN 1,0
585 SCREEN 1,0
715 IF INKEY$="" THEN GOTO 715
```

Include this line if you have a Spectrum (Timex 2000):

```
790 IF D=1 THEN DRAW NX-OX,NY-OY
```

Include these lines if you have an Apple:

```
 35 VTAB(21)
205 VTAB(21)
525 VTAB(21)
585 VTAB(21)
```

```
*10 gmode
 20 DIM A%(12):DIM L%(12): DIM D%(12):DIM C%(12)
*30 cls
*40 LET WI%=screen width
*50 LET HI%=screen height
 60 LET SX%=WI%/2
 70 LET SY%=HI%/2
 80 LET W=0.0174532925
 90 PRINT "TYPE 1 TO DRAW A SHAPE"
100 PRINT "TYPE 2 TO MAKE A PATTERN"
110 INPUT R%
*120 move SX%,SY%
130 LET OX%=SX%
140 LET OY%=SY%
150 LET A%=90
*160 clg
170 IF R%=2 THEN GOSUB 510
180 IF R%=1 THEN GOSUB 200
190 GOTO 90
*200 cls
210 LET I%=1
220 LET S=1
230 LET B%=A%
240 LET D%=1
250 LET C%=1
260 LET L%=WI%/20
270 LET A%=A%+150
280 FOR H%=1 TO 3
290 GOSUB 730
300 LET A%=A%+120
310 NEXT H%
320 LET A%=B%
330 LET D%=1
340 INPUT "ANGLE TO TURN";AA%
350 INPUT "LENGTH OF MOVE";L%
360 INPUT "DRAW (Y/N)";D$
370 IF D$="N" THEN LET D%=0:GOTO 390
380 INPUT "DRAWING COLOUR";C%
```

Leave out the % signs if you have a Spectrum (Timex 2000) or a Dragon.

Computer moves cursor to centre of screen. (SX and SY are the screen width and height divided by two (see lines 60 and 70).

Computer comes here if you type 1 in line 90.

These lines set the size of the "turtle" pointer. At line 290, the computer goes to the subroutine in line 730 to draw the turtle on the screen.

If you have an Apple, replace H% with H in lines 280 to 310.

Questions asked when drawing the shape. If you are using a BBC, replace the semi-colons in lines 340-360 and line 380 with commas.

The program listing continues over the page.

```
390 LET A%=A%-AA%
400 IF AA%>360 THEN LET AA%=AA%-360
410 IF AA%<0 THEN LET AA%=AA%+360
420 LET A%(I%)=A%
430 LET L%(I%)=L%
440 LET D%(I%)=D%
450 LET C%(I%)=C%
460 GOSUB 730
470 IF I%=12 THEN RETURN
480 LET I%=I%+1
490 IF L%>0 THEN GOTO 230
500 RETURN
510 LET P%=0
*520 cls
*530 clg
540 INPUT "STARTING SCALE";S
550 INPUT "NUMBER OF STEPS";NS%
560 INPUT "SCALE CHANGE";ST
570 INPUT "ANGLE CHANGE";Q%
*580 cls
590 FOR X%=1 TO NS%
600 LET OX%=SX%
610 LET OY%=SY%
620 FOR T%=1 TO I%
630 LET A%=A%(T%)+P%
640 LET L%=L%(T%)*S
650 LET D%=D%(T%)
660 LET C%=C%(T%)
670 GOSUB 730
680 NEXT T%
690 LET S=S+ST
700 LET P%=P%+Q%
710 NEXT X%
720 RETURN
730 LET NX%=OX%+(L%*COS(W*A%))
740 LET NY%=OY%+(L%*SIN(W*A%))
750 IF NX%<0 OR NX%>WI% THEN GOSUB 840:GOTO 830
760 IF NY%<0 OR NY%>HI% THEN GOSUB 840:GOTO 830
*770 gcolour C%
*780 move OX%,OY%
*790 IF D%=1 THEN draw NX%,NY%
*800 IF D%=0 THEN move NX%,NY%
810 LET OX%=NX%
820 LET OY%=NY%
830 RETURN
*840 beep
850 IF R%=2 THEN LET X%=NS%:LET T%=I%
860 IF R%=1 THEN LET I%=I%-1:LET A%=A%+AA%
870 RETURN
```

This line tests whether 0 has been typed in for LENGTH OF MOVE in line 350. If so, the program goes back to line 190 and then to line 90 and asks the question in that line.

Computer comes here if you type 2 in line 100.

Questions asked to make the pattern. If you are using a BBC, replace the semi-colons in these lines with commas.

If you have an Apple, replace X% and T% with X and T in lines 590 to 710.

If you have an Apple, replace X% with X and T% with T in this line.

Bouncing bug

In this game you control a bouncing bug which moves around devouring weeds which have infested your garden. As it stands, the program uses ordinary keyboard characters to represent a bug (X) and a weed (#). You can make up your own shapes for the bugs and weeds if your computer allows you to. There are examples of how to do this on a BBC or Spectrum (Timex 2000) on page 43. You can find out more about making your own character shapes by looking in your manual under user-defined characters or user-defined graphics. (You cannot define your own characters on a Dragon or Apple.)

How to play Bouncing Bug

When you run this program, your garden is drawn on the screen, gradually filling with more and more weeds. Press the space bar to start the game. Your first bug appears and you can move it up, down, left and right by pressing the "A", "Z", "," and "." keys. It also bounces off the walls of the garden. When it hits a weed, the weed disappears. You get a score for each weed eaten. If a bug eats all the weeds before the time runs out, you also get a bonus score. If not, you lose that bug and another bug appears on the screen. In each game you have three bugs. The time allowed for eating weeds decreases with each bug.

There is an invisible hole in the garden. If the bug falls down the hole you lose it. The hole is in a different place each time.

When you have lost all three bugs, the computer asks you if you want to play again and you type Y or N.

```
 10 LET WI%=screen width
 20 LET HI%=screen height
 30 LET V%=1
 40 LET W%=0
 50 LET G%=15:LET Z=1
 60 LET TS%=0
 70 LET NB%=3
 80 DIM X%(G%):DIM Y%(G%)
 90 GOSUB 830
100 GOSUB 550
110 GOSUB 800
120 beep:GOTO 270
130 LET Q$=inkey$
140 IF Q$="A" THEN LET W%=-1:LET V%=0
150 IF Q$="Z" THEN LET W%=1: LET V%=0
160 IF Q$="," THEN LET V%=-1: LET W%=0
170 IF Q$="." THEN LET V%=1:LET W%=0
180 GOSUB 440
190 print tab(X%,Y%);" "
200 LET X%=X%+V%
210 LET Y%=Y%+W%
220 IF X%<1 THEN LET X%=1:LET V%=-V%
230 IF Y%<4 THEN LET Y%=4:LET W%=-W%
240 IF X%>WI%-1 THEN LET X%=WI%-1:LET V%=-V%
250 IF Y%>HI%-2 THEN LET Y%=HI%-2:LET W%=-W%
260 print tab(X%,Y%);CHR$(D%)
270 FOR I%=1 TO G%
280 IF X%=X%(I%) AND Y%=Y%(I%) THEN GOSUB 490
290 NEXT I%
300 IF X%=P% AND Y%=Q% THEN beep:LET T%=1
310 print tab(X%,Y%);CHR$(88)
320 LET T%=T%-1
330 print tab(0,2);"SCORE ";TS%;" TIME ";T%;"    "
340 IF T%>0 THEN GOTO 130
350 IF B%>0 THEN print tab(5,5);"BONUS ";B%
```

Leave out all the % signs if you have a Dragon or a Spectrum (Timex 2000).

If you have a Spectrum (Timex 2000), type in "a", "z", "n" and "m" instead of "A", "Z", "," and "." in lines 140 to 170.

This program uses ordinary text mode instead of a high resolution graphics mode, so type the following numbers into lines 10 and 20 for screen width and height instead of the numbers in the conversion chart:

Spectrum (Timex 2000)
Screen width: 31
Screen height: 21

Dragon
Screen width: 31
Screen height: 15

Apple
Screen width: 39
Screen height: 22

BBC
Add either of these sets of lines to the program telling it which mode to use, and the corresponding screen dimensions:

```
 5 MODE1
10 LET WI%=39
20 LET HI%=30

 5 MODE5
10 LET WI%=19
20 LET HI%=30
```

If you have an Apple, replace I% with I in lines 270 to 290, and C% with C in line 360 over the page.

The program listing continues over the page.

```
 360 FOR C%=1 TO 5000:NEXT C%
 370 LET TS%=TS%+B%
 380 IF B%=0 THEN LET NB%=NB%-1
 390 IF NB%>0 THEN GOTO 100
*400 print tab(5,5);"PLAY AGAIN ? (Y/N)"
 410 INPUT Q$
 420 IF Q$="Y" THEN RUN
 430 STOP
 440 IF W%=1 THEN LET D%=86
 450 IF W%=-1 THEN LET D%=94
 460 IF V%=-1 THEN LET D%=60
 470 IF V%=1 THEN LET D%=62
 480 RETURN
 490 LET S%=S%+1
 500 LET TS%=TS%+5
 510 LET X%(I%)=0
 520 LET Y%(I%)=0
 530 IF S%=G% THEN LET B%=T%/3:LET T%=1
 540 RETURN
*550 cls CLS
*560 colour 3 COLOUR3
*570 print tab(0,1);"BOUNCING BUG : BUGS ";NB% PRINT TAB(0,1);"Bouncing bug: Bugs"
 580 LET S%=0
 590 LET B%=0
 600 LET Z=Z+0.3
 610 LET T%=1000/Z
 620 FOR I%=1 TO G%
*630 colour INT(rnd(1)*2+1)
 640 GOSUB 800
 650 LET X%(I%)=X%:LET Y%(I%)=Y%
*660 print tab(X%(I%),Y%(I%));CHR$(35)
 670 NEXT I%
 680 GOSUB 800
 690 LET P%=X%:LET Q%=Y%
*700 colour INT(rnd(1)*2+1)
 710 FOR I%=1 TO WI%
*720 print tab(I%,3);"*"
*730 print tab(I%,HI%-1);"*"
 740 NEXT I%
 750 FOR I%=3 TO HI%-1
*760 print tab(0,I%);"*"
*770 print tab(WI%,I%);"*"
 780 NEXT I%
 790 RETURN
*800 LET X%=rnd(1)*(WI%-1)+1
*810 LET Y%=rnd(1)*(HI%-5)+4
 820 RETURN
 830 GOSUB 550
 840 GOSUB 620
*850 colour 3
*860 print tab(0,2);"PRESS SPACE TO START"
*870 LET Q$=inkey$
 880 IF Q$<>" " THEN GOTO 840
 890 RETURN
```

If you have a Dragon or a Spectrum (Timex 2000), replace the number 5000 with 500 in line 360.

If you have an Apple, replace I% with I in lines 510 and 520.

If you have an Apple, replace I% with I in lines 710 to 780.

Extra conversions for the Spectrum (Timex 2000) and Dragon

If you have a Spectrum or a Dragon, include these lines in the program:

```
 530 IF S=G THEN LET B=INT(T/3):LET T=1
 610 LET T=INT(1000/Z)
*800 LET X=INT(rnd(1)*(WI-1)+1)
*810 LET Y=INT(rnd(1)*(HI-5)+4)
```

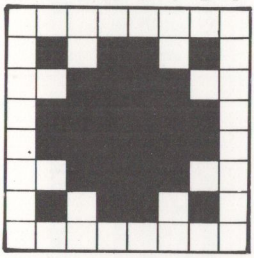
If you have a Dragon, include these lines in addition to those above:

```
 560 REM
 630 REM
 700 K=INT(RND(0)*2)*16+32
 720,730,760,770 Replace "*" with CHR$(134+K);
 850 REM
```

User-defined characters for the BBC and Spectrum (Timex 2000)

User-defined characters are letter-sized shapes which you design yourself and then program the computer to display on the screen. At the bottom of the page are some program lines for the BBC and Spectrum to include in the Bouncing Bug program. They make the user-defined characters for the bugs and weeds shown below.

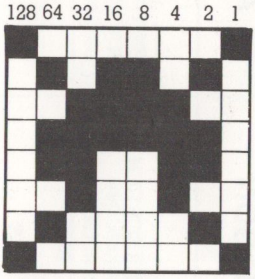
128 64 32 16 8 4 2 1



Bug

0
2+8+16+64=**90**
4+8+16+32=**60**
2+4+8+16+32+64=**126**
2+4+8+16+32+64=**126**
4+8+16+32=**60**
2+8+16+64=**90**
0

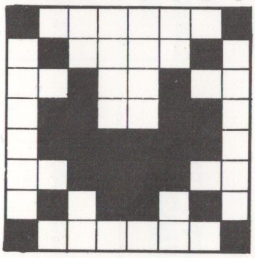
128 64 32 16 8 4 2 1



Bug moving down

1+128=**129**
2+8+16+64=**90**
4+8+16+32=**60**
2+4+8+16+32+64=**126**
2+4+32+64=**102**
4+32=**36**
2+64=**66**
1+128=**129**

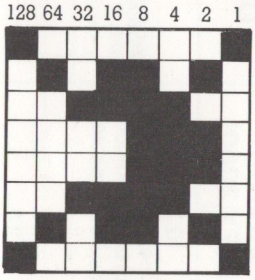
128 64 32 16 8 4 2 1



Bug moving up

1+128=**129**
2+64=**66**
4+32=**36**
2+4+32+64=**102**
2+4+8+16+32+64=**126**
4+8+16+32=**60**
2+8+16+64=**90**
1+128=**129**

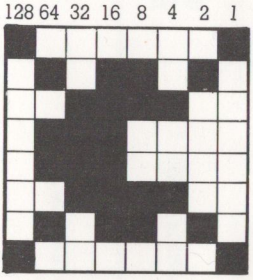
128 64 32 16 8 4 2 1



Bug moving left

1+128=**129**
2+8+16+64=**90**
4+8+16+32=**60**
2+4+8=**14**
2+4+8=**14**
4+8+16+32=**60**
2+8+16+64=**90**
1+128=**129**

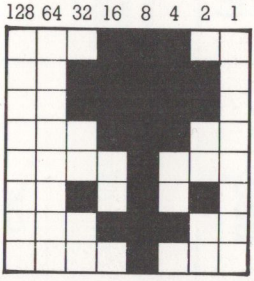
128 64 32 16 8 4 2 1



Bug moving right

1+128=**129**
2+8+16+64=**90**
4+8+16+32=**60**
16+32+64=**112**
16+32+64=**112**
4+8+16+32=**60**
2+8+16+64=**90**
1+128=**129**

128 64 32 16 8 4 2 1



Weed

4+8+16=**28**
2+4+8+16+32=**62**
2+4+8+16+32=**62**
4+8+16=**28**
8
2+8+32=**42**
4+8+16=**28**
8

To design a character, you draw an 8 × 8 grid like the ones above and fill in the squares to make your shape. Then you number the columns of squares as shown. You feed the information for a character into the computer row by row, by adding together the numbers at the tops of the columns where you have shaded in squares. (You can see the numbers for these shapes in the program lines below. You can design your own shape and change the numbers if you like.)

Program lines for the Spectrum (Timex 2000)

```
  85 GOSUB 900
 310 PRINT AT Y,X;CHR$(144)
 440 IF W=1 THEN LET D=145
 450 IF W=-1 THEN LET D=146
 460 IF V=-1 THEN LET D=147
 470 IF V=1 THEN LET D=148
 660 PRINT AT Y(I),X(I);CHR$(149)
 900 FOR I=0 TO 47
 910 READ A
 920 POKE USR "A"+I,A
 930 NEXT I
 940 RETURN
 950 DATA 0,90,60,126,126,60,90,0
 960 DATA 129,90,60,126,102,36,66,129
 970 DATA 129,66,36,102,126,60,90,129
 980 DATA 129,90,60,14,14,60,90,129
 990 DATA 129,90,60,112,112,60,90,129
1000 DATA 28,62,62,28,8,42,28,8
```

Program lines for the BBC

```
 85 GOSUB 900
310 PRINT TAB(X%,Y%);CHR$(224)
440 IF W%=1 THEN LET D%=225
450 IF W%=-1 THEN LET D%=226
460 IF V%=-1 THEN LET D%=227
470 IF V%=1 THEN LET D%=228
660 PRINT TAB(X%(I%),Y%(I%));CHR$(229);
900 VDU 23,224,0,90,60,126,126,60,90,0
910 VDU 23,225,129,90,60,126,102,36,66,129
920 VDU 23,226,129,66,36,102,126,60,90,129
930 VDU 23,227,129,90,60,14,14,60,90,129
940 VDU 23,228,129,90,60,112,112,60,90,129
950 VDU 23,229,28,62,62,28,8,42,28,8
960 RETURN
```


*Different versions of the bug are displayed according to the direction in which it is moving to give the impression of animation. See *Animation and colour-swapping* on page 13.

Conversion chart

This chart shows you how to convert the pseudo-commands in the program listings into commands which will work on your computer. Look along the line in the chart in which the pseudo-command is printed until you reach the command printed in your computer's column.

Replace the letters X, Y and N which occur after some of the commands in the chart with the letters or numbers printed after the pseudo-commands in the program listings.

Pseudo-commands used in program listings	Description of command	BBC/ARC	SPECTRUM (TIMEX 2000)	DRAGON	APPLE
gmode	Set the graphics mode. (You need a mode with at least four colours.)	MODE 1	REM (Computer automatically draws in high resolution graphics. See footnote 1.)	PMODE 3:SCREEN 1,0	HGR
cls	Clear the text screen.	CLS	CLS	CLS	HOME
clg	Clear the graphics screen.	CLG	CLS	PCLS	HGR
gcolour N	Set the colour for drawing lines.	GCOL 0,N	INK N	COLOR N,1	HCOLOR=N
move X,Y	Move the cursor to point X,Y	MOVE X,Y	PLOT OVER 1; INVERSE 1;X,Y	LINE(X,Y)−(X,Y),PRESET	HPLOT X,Y
draw X,Y	Draw a line from last cursor position to new position X,Y	DRAW X,Y	DRAW X,Y	LINE −(X,Y),PSET	HPLOT TO X,Y
dot X,Y	Plot a point at position X,Y	PLOT 69,X,Y	PLOT X,Y	LINE (X,Y)−(X,Y),PSET	HPLOT X,Y
print tab(X,Y);	Print text or variable at position X,Y	PRINT TAB(X,Y);	PRINT AT Y,X; (See footnote 2.)	PRINT @ 32*Y+X, "_____"; (See footnote 2.)	VTAB(Y+1):HTAB(X+1): PRINT "_____" (See footnote 2.)
beep	Make a short beep noise	VDU 7	BEEP 1,1	SOUND 90,1	PRINT CHR$(7);
X$=inkey$	Test to see if a key on the keyboard has been pressed.	X$=INKEY$(0)	X$=INKEY$	X$=INKEY$	X$="":IF PEEK(−16384) >127 THEN GET X$
rnd(1)	Pick a random number between 0 and 0.9999	RND(1)	RND	RND(0)	RND(1)
colour N	Set text colour for PRINT statements.	COLOUR N	INK N	REM (Computer cannot print text in colour. See footnote 1.)	REM (Computer cannot print text in colour in this resolution. See footnote 1.)
screen width	Width of screen you can draw in	1279	255	255	279
screen height	Height of screen you can draw in	1023	175	191	159

1. Computer ignores the word REM and carries on to the next line. It is safest to keep the line in the program using REM rather than leave it out altogether, in case there is a GOTO elsewhere in the program telling the computer to go to that line.

2. Remember to change round the numbers represented here by X and Y so that the number for Y comes first.

Graphics words

Animation Moving pictures on a screen, or movement of characters within a picture.

Audio-visual Information presented in the form of films or slides with a spoken or taped commentary.

Caption generator A special computer which can create text in different styles, sizes and colours. It is used in film and television titles and credits, for example.

Cathode ray tube (CRT) The device in television sets and monitors which converts electrical signals into pictures or words on the screen.

Character generator A program in the computer's memory that tells it what shapes to light up on the screen to make the characters (letters, numbers and symbols) on the computer's keyboard.

Clipping Removing parts of a screen picture.

Colour-swapping The process by which everything in one colour on the screen is changed into another colour.

Computer-aided design (CAD) Using computer programs and graphics equipment to help in design work.

Depth clueing Altering the shade of a colour according to how far away it is in a picture, to give the impression of distance.

Digital frame store See **Graphics page.**

Digitizing camera A special camera that converts information about a picture into numbers and feeds it into a computer, which can then display the picture on the screen.

Digitizing pad A touch-sensitive pad connected to a computer which registers the movement of a pen on the pad and displays the shape of the drawing on the screen. (See also **Graphics tablet.**)

Display processor A chip inside a computer for controlling the screen display.

Film plotter A device that transfers picture information from a computer on to film. It uses lasers or fibre optics to copy the picture from the computer's memory on to film.

Frame buffer See **Graphics page.**

Graphics characters Ready-made shapes, for example, circles and squares, which can be displayed on the screen by pressing certain keys. They are stored in the computer's memory in the same way as ordinary letters, numbers and symbols.

Graphics modes Different ways of organizing the screen display. Some computers have several graphics modes in which they can produce pictures of different quality, or resolution, with different numbers of colours.

Graphics page or **Frame buffer** This is the part of a microcomputer's memory that stores the information for a picture. The information is in the form of numbers representing the colour and brightness of each pixel. In larger computers it is called the digital frame store.

Graphics tablet A touch-sensitive pad on which you draw with a pen. (See **Digitizing pad.**) Alternatively, a pad over which the pen is supported in a jointed arm. As you draw, the computer works out the position of the pen from the angles in the joints and displays the drawing on the screen.

Graphics terminal Screen and keyboard with a light pen or graphics tablet, connected to a main central computer.

Hidden line removal Erasing the lines in a picture that go behind an object, so that the object appears solid.

High resolution A screen display made up of very small pixels so the lines in the pictures are thin and the curves are smooth.

Image enhancement Clarification of blurred or indistinct pictures, for example, photographs sent back from space probes to base. Photographs are fed into a computer which compares each pixel with its neighbour. It works out which pixels are likely to be genuine details and which are the result of interference. The computer also increases the contrast and intensifies colours.

In-betweening The process whereby a computer works out and draws shapes which show one object changing into another, or a character moving from one position to another (see **Key frames**).

Joystick Hand-held, moveable stick in a firm base connected to a computer. You move the stick to control mobile characters on the screen, or draw lines, according to the computer's program.

Key frames Pictures drawn by an artist on a computer showing, for example, a character in two different positions. The computer then works out the stages which show the character moving from the first position to the second (see **In-betweening**).

Light pen Light sensitive pen used for drawing pictures directly on the screen.

LOGO A programming language used for designing shapes and patterns.

Low resolution Pictures made up of few, large pixels, so lines are thick and curves made up of steps.

Monitor Specially designed screen that understands the same signals as a computer.

Monochrome Black and white screen display.

Motion control Computerized control of film cameras.

Mouse Hand-held device with wheels which is connected to a computer. As the mouse is moved, it feeds information about its direction back to the computer which draws a corresponding line on the screen.

Palette The set of colours available to the computer for use on the screen.

Pixel Small areas into which the computer divides the screen. The computer can turn each pixel on or off and make them different colours to build up pictures.

Playfield Area on a computer screen reserved for the main action in a computer game.

Plotter Device attached to a computer which draws out computer graphics using a pen supported over a sheet of paper.

Printer Device attached to a computer which prints out text. Some printers can also print out graphics using tiny dots.

Raster scanning Method of display used by televisions and most monitors. An electron beam is fired through a cathode ray tube (see above). The beam scans across and down the back of the screen, lighting up certain pixels on its way to make up a picture.

Real-time Direct computer response to user's instructions; for example, in flight simulators.

RGB screen The letters stand for red, green and blue and refer to a colour screen in which all the colours are made up of different combinations of red, green and blue dots.

Simulation Computerized imitation of real events, used in training or scientific experiments, for example.

Teletext Computerized information sent to TV sets along with broadcast TV signals.

Text mode A kind of screen display that can only show text and not graphics. (See **Graphics modes.**)

Translation Changing a picture on a screen to give the impression that the viewer has changed position.

Turtle graphics Shapes and patterns on a screen made by moving a small pointer, or turtle, around. The programming language LOGO uses turtle graphics.

Vector graphics Instead of the electron beam scanning across and down the screen as in raster scanning (see above), the beam is directed only where lines need to be drawn. It can move in any direction on the screen, lighting up pixels as it goes.

Videotex General name for information sent from central computerized databases to TV sets via the telephone lines or as broadcast TV signals.

Viewdata Information sent from computerized databases to TV sets down telephone lines. Users can send messages back to the main computer in the same way.

Window Special area on the screen set aside for text or graphics alone.

Wire-frame Line drawing of a three-dimensional object showing all the lines that make up the shape, including those that go round the back of it.

Index

3-D graphics, 14-15, 18, 19, 20, 21, 31
analogue, 10
  signals, 6
  to digital converter, 10
animation, 3, 13, 24, 26-27, 46
Apple computer, 3, 32, 33, 41, 44-45
arcade games 3, 12-13
art, computer, 3, 30-31
Atari computer, 9, 12
audio-visual, 28, 46
BASIC, 5
BBC computer, 3, 9, 12, 29, 32, 33, 37, 39, 41, 43, 44-45
binary, 5, 6, 8
bits, 5, 8, 9
Bitstik, 17
Bouncing Bug program, 41-43, 33
Bueno, Luz, 31
business graphics, 28
byte, 8, 9
CAD see computer-aided design
camera, 11, 28, 30
caption generator, 25, 46
cartoon animation, 3, 26-27
cathode ray tube (CRT), 6, 46
central processing unit (CPU), 4
character generator, 5, 7, 46
charts, 25, 28
chips, 7
  display processor, 12, 46
  graphics, 7, 12
chromakey, 23
clipping, 14, 46
colour-swapping, 13, 46
Commodore 64 computer, 12
compiler, 5, 19
computed tomography, 20
computer-aided design (CAD), 16-17, 46
  equipment, 16-17
  programs, 16-17, 20, 21
Conversion Chart, 32, 44-45
CPU see central processing unit
CRT see cathode ray tube
depth clueing, 15, 46
digital, 10
  frame store, 22, 46
  signals, 6
  wave shape, 6
digitizing camera, 21, 22, 27, 31, 46
digitizing pad, 11, 46
disk drive, 17
display processor, 12, 46
Dragon computer, 3, 34, 33, 34, 36, 37, 39, 41, 42, 44-45
electron beam, 6, 11, 46, 47
fibre optics, 11, 20
film plotter, 11, 46
films, 3, 24, 26-27
  special effects, 22-23
Flair, 24
floppy disk, 17, 25, 28
frame buffer, 4, 46
games, 12
  Bouncing Bug, 41-43, 33
graphics, 10
  characters, 7, 9, 46
  commands, 10, 44-45
  modes, 7, 9, 46
  page, 4, 5, 6, 12, 22, 46
  programs, 4-5
  programs to run, 3, 5, 32-43
  tablet, 11, 16, 24, 26, 30, 31, 46
terminal, 16, 46
graphs, 25, 28
hardware, 4
heat pictures, 20
hidden line removal, 15, 17, 47
high resolution, 7, 8, 9, 24, 47
home computers, 3, 9, 10, 12, 14, 15, 16, 18, 19, 25, 28
IBM, 30
image enhancement, 19, 47
in-betweening, 26-27, 47
  program, 33, 36-37
integer variables, 33
interpreter, 4, 5, 19
joysticks, 11, 17, 47
K, see kilobyte
key frames, 26, 27, 47
keyboard, 4, 7, 10, 16
kilobyte, (K), 8, 9
lasers, 11, 20
library, 24, 27
light pen, 11, 16, 21, 26, 27, 30, 47
LOGO, 5, 47
low resolution, 7, 8, 47
machine code, 4, 5
mainframe computers, 22
maps, 25
matting, 23
memory, 4, 6, 7, 8, 12, 22, 29, 33
  location, 6
  size, 8, 9
Metaloom, 30
minicomputers, 22, 30
modes, 7, 9, 29, 46, 47
monitor, 6, 8, 10, 31, 47
monochrome, 10, 47
motion control, 23, 47
mouse, 10, 47
NASA, 14, 19
Paint Box, 24
palette, 13, 24, 47
Pattern Generator program, 32, 34-35
pixels, 4, 5, 6, 7, 8, 9, 10, 19, 21, 29, 31, 47
playfield, 12, 47
plotter, 11, 47
printers, 10, 31, 47
program, 4, 10, 11, 14, 15, 20, 21, 24, 27, 28
  programs to run, 3, 32-43
programming languages, 5, 30
RAM, 4, 5, 6, 7, 8, 9
random access memory, see RAM
raster scanning, 6, 47
read only memory, see ROM
real-time, 18, 47
Resch, Ronald, 31
resolution, 7, 8, 9, 22, 24, 31, 47
RGB, 10, 47
ROM, 4, 5, 7
scrolling, 25
sculpture, computer-assisted, 31
simulation, 18-19, 20, 23, 47
  astronomy, 19
  flight, 18
  home computer, 18, 19
  scientific, 19
  space, 19
Smullin, Frank, 31
software, 4, 24
Space Shuttle, 16
special effects, 22-23, 24
Spectrum, 3, 7, 32, 33, 34, 35, 36, 37, 39, 41, 42, 43, 44-45
Star Wars, 23
stylus, 11, 16, 24
surgery, computer-aided, 20
Teletext, 29, 47
television, see TV
Telidon, 29
Timex 1000, see ZX81
Timex 2000, see Spectrum
translation, 14, 47
Tron, 22, 23
turtle, 38, 47
Turtle Graphics program, 32, 38-40
TV, 6, 8, 10, 11, 29
  graphics, 24
  how it works, 6
  special effects, 22
user-defined characters, 41, 43
user-defined graphics, see user-defined characters
variables, 33
vector graphics, 6, 47
Vectrex games module, 6
VIC 20, 7, 9, 12
Videotex, 29, 47
Viewdata, 29, 47
Voyager, 19
Walt Disney, 22
weaving pictures, 30
windows, 12, 47
  graphics, 12
  text, 12
wire frame, 14, 15, 17, 47
X-ray, 20
ZX81, 7

Picture on page 21 © CBA Inc/New Scientist. Pictures on page 31 © Luz Bueno, © Frank Smullin and © Ronald Resch.

First published in 1983 by Usborne Publishing Ltd., 20 Garrick Street, London, WC2E 9BJ, England.

© 1983 Usborne Publishing
The name Usborne and the device are Trade Marks of Usborne Publishing Ltd.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the publisher.

HP9845B conversions

Gmode	Plotter is 16, Graphics Graphics
Cls	Print Chr$(12)
Clg	Clear

Usborne Computer Books

Usborne Computer Books are colourful, straightforward and easy-to-understand guides to the world of home computing for beginners of all ages.

Usborne Guide to Computers A colourful introduction to the world of computers. *"Without question the best general introduction to computing I have ever seen."* Personal Computer World

Understanding the Micro A beginner's guide to microcomputers, how to use them and how they work. *"This introduction to the subject seems to get everything right."* Guardian

Computer Programming A simple introduction to BASIC for absolute beginners. *"... lucid and entertaining ..."* Guardian

Computer and Video Games All about electronic games and how they work, with expert's tips on how to win. *"The ideal book to convert the arcade games freak to real computing."* Computing Today

Computer Spacegames, Computer Battlegames Listings to run on the ZX81, Spectrum, BBC, TRS-80, Apple, VIC 20 and PET. *"Highly recommended to anyone of any age."* Computing Today

Practical Things to do with a Microcomputer Lots of programs to run and a robot to build which will work with most micros.

Computer Jargon An illustrated guide to all the jargon.

Computer Graphics Superbly illustrated introduction to computer graphics with programs and a graphics conversion chart for most micros.

Write Your Own Adventure Programs Step-by-step guide to writing adventure games programs, with lots of expert's tips.

Machine Code for Beginners A really simple introduction to machine code for the Z80 and 6502.

Better BASIC A beginner's guide to writing programs in BASIC.

Inside the Chip A simple and colourful account of how the chip works and what it can do.

+003·95

ISBN 0-86020-740-4

00395

9 780860 207405

Published in Canada by Hayes Publishing Ltd, 3312 Mainway, Burlington, Ontario, Canada, L7M 1A7.

Published in the USA by EDC PUBLISHING, 8141 E. 44th Street, Tulsa, Oklahoma 74145, USA.

ISBN 0 86020 740 4

£3.95

Printed in Spain

10 11 12/83